In Sync over Distance: Flexible Coordination through Communication in Geographically Distributed Software Development Work

by

Hyun Gyung Im

B.A. Social studies, Seoul National University, 1994
M.A. Sociology, Seoul National University, 1996

SUBMITTED TO THE PROGRAM IN SCIENCE, TECHNOLOGY AND SOCIETY IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
IN THE DEPARTMENT OF HISTORY, ANTHROPOLOGY, SCIENCE, TECHNOLOGY AND SOCIETY
AT THE MASSACHUSETTS INSTITUTE OF TECHNOLOGY
SEPTEMBER 2006

Signature of Author: _____
Program in Science, Technology and Society
September 6, 2006

Certified by: _____
JoAnne Yates
Sloan Distinguished Professor of Management, Sloan School of Management

Certified by: _____
Wanda J. Orlikowski
Eaton Peabody Professor of Communication Sciences, Sloan School of Management

Certified by: _____
Susan S. Silbey
Professor of Anthropology

Accepted by: _____
David A. Mindell
Dibner Professor of the History of Engineering and Manufacturing
Director, Program in Science, Technology and Society
Director of Graduate Studies, History, Anthropology and STS

In Sync over Distance: Flexible Coordination through Communication in Geographically Distributed Software Development Work

by

Hyun Gyung Im

Submitted to the Program in Science, Technology, and Society on September 6, 2006 in partial fulfillment of the requirements for the degree of Doctor of Philosophy in the Department of History, Anthropology, Science, Technology and Society

ABSTRACT

In this dissertation, I examine how the members of a distributed software development team (LC) operating entirely virtually for four and half years developed useful social practices to collaborate across time and space. Based on various communication data from LC, I analyze the communicative structuring of distributed work in members' daily practices. I show that "temporal flexibility," often mentioned as key advantage of virtual organizing, is socially accomplished through "boundary management," as members negotiate different temporal boundaries and learn and adapt to others' temporal patterns. Second, I identify dynamic coordination practices in LC that interweave multiple modes of communication and coordination in evolving work contexts, and demonstrate how these coordination practices facilitate temporal flexibility in LC. Finally, I analyze how members used the asynchronous communication medium of email to coordinate their tasks, using the notion of genre and genre system. My analysis suggests that communicating, coordinating, and temporal structuring are not distinctive activities, but are closely bound up with each other in a local practice; time, communication, and coordination are dynamically reconfigured over time, reflecting evolving work, social relations, and local contexts.

Key Words: distribute teams, virtual teams, virtual organizing, technology-mediated communication, temporal flexibility, coordinating, communicating, temporal structuring, social practices, communicative structuring, genre and genre system, reconfiguration of time, communication, and coordination.

Thesis supervisor: JoAnne Yates
Title: Sloan Distinguished Professor of Management, Sloan School of Management

To my grandmother,
李末述

Acknowledgement

This dissertation would never have been completed without the tremendous support of so many colleagues, friends, and family. It is almost impossible to express in words my sincere gratitude for their time, insights, interest, attention, friendship, faith, and love.

I am deeply indebted to my committee-Wanda Orlikowski, Susan Silbey, and JoAnne Yates. They stayed with me through long periods of silence, pushed me to go on, and helped me work out a realistic plan to get this dissertation done. They guided me through difficult choices, challenged me in encouraging ways, and provided me with critical intellectual insights. I am very grateful to JoAnne Yates, my advisor for her endless support and encouragement; she has been an invaluable mentor, helping me find my way through both academic and personal issues: topics of JoAnne's advice ranged from theory and research to the choice of a sleep aid and bedside novels. I am also very grateful to Wanda Orlikowski; it was the doctoral seminar she taught in my second semester at MIT that introduced me to the field of organizational studies. Wanda also introduced me to the SeeIT (Social and Economic Explorations of Information Technology) Project of which she was the principal investigator and JoAnne one of the co-principal investigators. I learned from the collaboration with JoAnne and Wanda as much as I learned from each of them individually. It was great honor and pleasure to work with them. I was very fortunate to work with Susan Silbey who joined my committee later in the dissertation process and has been wonderful influence ever since. I deeply thank Susan for her keen eye to find connections between various domains of theories and social experience, her encouragement to grow from being "meticulous" to making a "bold" statement, and her constant reminder that the best dissertation is a done dissertation. My committee members' passion for research and life will continue to be a source of inspiration.

My colleagues from the SeeIT Project supplied a steady stream of intellectual discussion. I very much enjoyed learning from colleagues working on other sub-projects, Joao Cunha, Kate Kellogg, Maureen Scully, and Bin Zhu. I would especially like to

express my deep appreciation to those who worked with me on the LC project at various stages of the project. Nils Fonstad offered to do the inter-rater reliability test of the very first coding scheme of LC's email communication, and in doing so, helped me elaborate and clarify the coding categories. I also thank students who coded various parts of the communication data, Scott Ostler, Changwoo Do, Xu Song Xie, Heng Xu. The collaboration with Heng-Xu and Tanu Gosh has been invaluable; we shared not just the data but also ideas, insights, laughter and sighs. Their feedback, support, and encouragement kept me sane, making the most tedious part of my research enjoyable. I am most grateful to our project manager, Stephanie Woerner for virtually unlimited support; she helped me at every stage of dissertation project-conception of ideas, data collection and analysis, writing processes, various forms of presentations, and final editing. Stephanie brainstormed with me when I had writer's block, filled my stomach as well as soul with berries from her back yard, and endured my sluggish progress with incredible patience.

I would like to thank the members of Little Company (LC) who generously provided the data and their time for the LC project. Some of the jokes they made in their emails made me laugh out loud in an otherwise quiet office. Special thanks to "Dan" for his continuous support, responding to my inquiries and reading every manuscript to check the confidentiality issues.

This dissertation was funded by the National Science Foundation under grant number IIS-0085725. I am also deeply grateful for the financial support from the program in Science, Technology, and Society at MIT.

There have been numerous colleagues, teachers, and friends who helped, supported, and pushed me in one way or another. I especially want to thank Deborah Fitzgerald who guided me through my first two years at MIT as my advisor. I thank Joe Dumit, Michael Fischer, and Hugh Gusterson for inspiring ideas and guidance, and Lotte Baylin and Pablo Boczkowski for their invaluable feedback on the ideas, presentations, and manuscripts that eventually became this dissertation. I am grateful to Wen-Hua Kuo, Hannah Landecker, Rachel Prentice, Aslihan Sanal, Heinrich Schwartz, for their advice on every aspect of graduate student life; Sang-Hyun Kim for rescuing me many times with conversations over a pint of beer and endless news of East-Asian STS communities;

and Chihyung Jeon for his calm support. Visiting scholars on my floor, especially Masamichi Takahashi and Hong Joo Lee never failed to cheer me up with humor, enthusiasm, and freshly brewed coffee. I also wish to thank the administrative staff at the program in STS who have been supportive in all respects, especially Chris Bates, Debbie Meinbresse, Shirin Fozi, Deb Fairchild, and Karen Gardner; I am most grateful to Kris Kipp who offered invaluable assistance, intervention, and unique encouragement.

Many thanks to my old friends who showed me friendship grows under many circumstances. My two best friends, Juyeon Sohn and Jihyun Jeon, always encouraged me to be myself during all these years we have been living on opposite sides of this continent. Another two best friends from high school, Yu-Kyung Park and En-Young Jeong always welcomed me in open arms, sheltered me in their homes, and shared their wisdom with me whenever I visited Seoul, Korea. I cannot thank enough to my dearest friend, Waty Lilaonitkul who has been always there through some good times and an awful lot of bad times; I am utterly grateful for the countless conversations we had, which helped me put things into perspective and come to peace with myself, even in the most spiritually and emotionally taxing situations.

My deepest appreciation and thanks to my family: to my sister, Engyung who quietly supported my endeavor by taking up the role of the eldest; to my brother, Dongkyu who always embraced me with the biggest heart; to my father Byunghwan Im who has the genuine curiosity, always on the lookout for new things to learn, and who encouraged me to do the same; and to my mother Junghee Ryu who taught me to stay hopeful even when life seems full of adversities. Without their unconditional love and support, none of this would have been possible.

Kooga, you provided unwavering support when I needed it most. You endured me in a most difficult time and gave me strength and courage to continue. For your companionship, tolerance, patience, care and love, I thank you with all my heart.

I dedicate this dissertation to my grandmother. She inspired her adolescent granddaughter to be courageous and independent, for which I am forever thankful. May she rest in peace.

Table of Contents

LIST OF FIGURES 10
LIST OF TABLES 11

INTRODUCTION 13
Literature Review *17*
Temporal structuring in practice 17
Distributed coordination and communication 22
Temporal coordination 27
Putting it together: Temporal structuring of distributed work through communication 30
Site and method *32*
Structure of the Dissertation *40*
PUTTING DISTRIBUTED WORK IN CONTEXT: LITTLE COMPANY (LC) 42
Team *42*
Work *50*
Technology *57*
ENACTING FLEXIBILITY IN PRACTICE 64
Flexibility: perception and practice *66*
How many and which hours did they work? *69*
Negotiated nature of flexibility *81*
Embracing flexibility: integrating temporal boundaries 81
Limiting flexibility: establishing temporal boundaries 86
Negotiating flexibility: flexibility as social accomplishment 90
Conclusion *101*
COORDINATING THROUGH COMMUNICATION OVER DISTANCE 104
Hidden challenges of distributed coordination *105*
Coupling coordination and communication *116*
Personal mode of coordination – dyadic communication 117
Group mode of coordination – group communication 124
Interweaving multiple modes of coordination and communication 129
Coordination, versatility, and flexibility *137*
Conclusion *144*
USING GENRES AND GENRE SYSTEMS FOR TEMPORAL COORDINATION 149
Communication genres and genre systems *149*
Genres used for Temporal Coordination *155*
Status report genre 155
Bug/error notification genre 158
Update notification genre 162
Interweaving genres in practice 164
Using genres over time: comparison between two phases *167*
Change in use pattern: fluid with task situations and contextual changes 167
Change in form: providing fixity in communication 171
Genre systems for temporal coordination *173*
Phone meeting management genre system 174
Group authoring genre system 182
Conclusion *194*
CONCLUSION 198

REFERENCES 209

APPENDIX A. CODING SCHEME I 220
APPENDIX B. CODING SCHEME II 222
APPENDIX C: DAN'S HOME OFFICE IN THE BASEMENT 225
APPENDIX D: ROBERT'S HOME OFFICE 227

List of Figures

Figure 1.1 LC 34
Figure 1.2 LC Timeline and data sources 35
Figure 1.3 Monthly distribution of LC emails 36
Figure 3.1 Daily temporal pattern in email communication (1996-2000) 74
Figure 3.2 Weekly temporal pattern in email communication (1996-2000) 75
Figure 3.3 Reasons for last minute adjustments 93
Figure 3.4 Meeting day 97
Figure 3.5 Change in weekly meeting time 98
Figure 4.1 Conflict around the Linker module 113
Figure 4.2 Percentage of email messages including disagreement among members 115
Figure 4.3 Dyadic emails between three full time members (1996-2000) 120
Figure 4.4 Dyadic phone between three full time members (1997-1999) 121
Figure 4.5 Attendance rate across four years 127
Figure 4.6 Communication frequency and code commits 133
Figure 4.7 Communication pattern over time (three full time members) 135
Figure 4.8 Phone use by individual member over time 136
Figure 4.9 Expertise of files committed by members 141
Figure 5.1 Distribution of email messages by sender 154
Figure 5.2 Timing and sequence of genres enacted by LC members on a work day 166
Figure 5.3 Genres in two phases 168
Figure 5.4 Genre use by three full-time members 169
Figure 5.5 The use of three genres in LC over time 169
Figure 5.6 Standardization of update notification in its form 172
Figure 5.7 Messages in meeting management genre system in two phases 179
Figure 5.8 Use of meeting management genre system by full-time members 180
Figure 5.9 Phone meetings and meeting management genre system in two phases 181
Figure 5.10 Use of group authoring system 191
Figure 5.11 Use of group authoring genre system by full time members 191
Figure 5.12 Monthly use of group authoring genre system in two phases 192

List of Tables

Table 1.1 Data sources 35
Table 1.2 Distribution of emails by sender and year (GMT) 36
Table 3.1 Main purpose of coordination through emails regarding phone meetings 93
Table 4.1 Email communication among LC members (1996-2000) 120
Table 4.2 Distribution of dyadic and group emails in LC 125
Table 4.3 LC member's attendance in phone meetings 127
Table 4.4 A work day in LC 132
Table 4.5 Correlations between combinations of communication and coordination types, 3 full time members 133
Table 5.1 Definition of genres and genre systems in LC 152
Table 5.2 Summary of data 154
Table 5.3 Genres enacted by LC members on a work day 165
Table 5.4 Group authoring genre system enacted around authoring an email 185
Table 5.5 Group authoring genre system enacted around authoring a press release 187
Table 5.6 Group authoring genre system around authoring presentation slides 188
Table 5.7 Summary of the group authoring genre system in two phases 193

CHAPTER 1
Introduction

One day in November 2000, a full-time software engineer in a small virtual start-up company sent the rest of his colleagues an email about a research project that proposed to study them as research subject:

> [This research project] plans to study groups like us. I've got email dating back to 21 feb 1997; [the manager of this project] thinks that this would be very interesting to them, and even more would be better. Phone records are something else that might be interesting to them.
>
> Is this something that we'd be willing to let them look at? Do you guys have e-mail going back even further? If we did it, they would want to interview us -- are we willing to talk about this? We would be a case study, but with the names changed to protect the guilty.
>
> I realize that this is a not a time when we have the time to make this decision, but we could think about it, or if either of you certainly know that we would not, then maybe we could relay that, too. If we have concerns (that would keep us from doing this) I think they would be interested in that, too.
>
> There's no money in it, but it is research.

The next morning, another full-time engineer replied, "I've got email going back to the beginning," implicitly giving his approval of being a "lab rat." After some internal discussion, they finally decided to share their data, experiences, memories and time for a case study of distributed teams, which became my dissertation project.

Distributed (or virtual) teams are groups of people with a common purpose who carry out interdependent tasks across locations and time through technological mediation (Lipnack and Stamps, 1997; Maznevski and Chudoba, 2000; Townsend et al., 1998). The possibilities of distributing work across time and space have been the subject of growing interest. Distributed work itself is not an entirely new phenomenon, however. History

provides ample examples that it preceded the arrival of modern technologies in transportation, communication, and information processing. For example, the Catholic Church two millennia ago and the Hudson Bay Company centuries ago both spanned a considerable part of the globe, and managing distributed work was one of their major challenges (King and Frost, 2002; O'Leary et al., 2002). Yet, it is undeniable that recent technological advances have radically shifted the scope and speed of distributed work, providing new means of organizing work across geographical, temporal, and cultural boundaries. Many companies began experimenting with various telecommuting programs decades ago. Road warriors have carried laptops into airplanes, hotel rooms, and clients' offices for many years. Work teams are increasingly composed of members who are spread across different cities or countries. For example, research and development laboratories are deployed in countries other than the home of their headquarters (Brockhoff, 1998) and software teams are composed of programmers from around the globe (Carmel, 1999). As traditional workplaces are transforming into virtual workspaces or "techno-territories" (Schwarz, 2003), the notion of working "any time, any place" has become a reality for many people.[1]

This rapid expansion of distributed work raises an important question: how do people working across different locations and time zones coordinate and align their activities? Despite the popular belief that technological advances would bring the "death of distance" (Cairncross, 1997), prior research has repeatedly reported that distance poses significant challenges to distributed teams. Distributed teams are observed to experience difficulties with insufficient context cues, feedback delays, and long interruptions in

[1] Some 137 million workers worldwide are estimated to be involved in various kinds of virtual work (Soloman, 2001).

communication that impede members' capability to interact, share information, and establish mutual understanding (Kraut et al., 1982; Sproull and Kiesler, 1986; Straus and McGrath, 1994). Virtual team members are often found to fail to share information about their local contexts or detect differences across locations, resulting in damage to coordination and interpersonal relationships (Cramton, 2001). Virtual teams also struggle to span multiple time zones to synchronize their efforts, and temporal coordination takes on heightened salience (Massey et al., 2002; Massey et al., 2003). However, there has been a paucity of empirical research on what virtual team members actually do to structure and coordinate their dispersed activities.

While learning much from the existing research on virtual teams (Lipnack and Stamps, 1997; Markus et al., 2000; Massey et al., 2002; 2003; Maznevski and Chudoba, 2000), I found it limited for two reasons. First, by framing "virtual" as a context that lacks qualities of the collocated environment (e.g., face-to-face interactions), many researchers focus primarily on the constraining aspects of technological mediation, leaving out the actual practices of distributed workers through which they create a shared context for collaboration. Second, much of the research is based on short-term teams whose work context and task differ significantly from those teams working on real-world projects (for exceptions, see Im et al., 2005; Panteli, 2004; Yoo and Alavi, 2004). As it takes time for team members to learn how to effectively collaborate in a distributed situation, short-term research might miss important team processes that can only be revealed through a long term observation, such as the very process of developing useful practices over time.

To understand what virtual team members actually do to structure and coordinate their dispersed activities, more longitudinal research on virtual teams in their "natural habitat" is needed. In this dissertation, I attempt such research in an empirical study of a virtual team that shifts focus to the actual practices of organizational members over an extended period of time. In other words, I report how the popular vision of working "anywhere and anytime" is materialized in a particular virtual team that operated almost entirely virtually over 4 years. In doing so, I pay special attention to the simultaneous reconfiguration of time, communication, and coordination in distributed work. How does organizing work in a dispersed way change the way people perceive, use, and manage time? How does it relate to the way they communicate and coordinate? To answer these questions, I examine the ideas and vision members brought with them when they set out to work in distributed mode, the strategies they used to actualize this vision, and how their everyday practices supported or conflicted with their vision. It will also bring to the fore the different local contexts in which members were embedded, the multiple structures members devised to organize themselves and their work, and the shared practices members developed to bridge the differences and stay connected. In short, I will examine how members actually worked together over a long period of time in a distributed context, and by doing so changed the context itself.

In the rest of this introductory chapter, I first review the literature and introduce my approach to research. Next, I describe the research site, data collection, and analysis methods. And then I close this chapter by briefly sketching the structure of the dissertation.

Literature Review

In this dissertation, I import ideas from multiple disciplines to understand the interrelations between time, communication, and coordination in technologically mediated distributed work.

Temporal structuring in practice

The notion of 'social time' indicates that sharing temporality is an important part of being a member of any human organization, whether it is a group, an organization, or a community or a society. From the outset, social science attempted to establish that time is above all a social phenomenon. For Durkheim (1976), who was concerned to theorize time from a thoroughly social perspective, time is a social category of thought and a product of society: "The rhythm of collective life dominates and encompasses the varied rhythms of all the elementary lives from which it results; consequently, the time that is expressed dominates and encompasses all particular durations" (1976, p. 69). While historically formed, Durkheim's social time is socially pregiven as a static conceptual reference system. As a result, even if it is collectively made and remade, it appears to the members of communities as a fixed and atemporal frame within which events and social actions are located and positioned.

This structural aspect of time as imposing patterns and order on social life is central in the functionalist approach to time (Coser and Coser, 1963; Sorokin and Merton, 1937; Zerubavel, 1979; 1981; 1985). In this approach, the need for social collaboration—i.e., to synchronize and coordinate social activities—is at the root of social systems of time. Zerubavel, following this line of thought, focused particularly on the systems of

temporal order that regulate and rigidify social action. For example, he argues that all social activities are conducted in accordance with schedules that specify their temporal location, duration, sequence, and rate of occurrence. According to him, these schedules represent the fundamental parameters of 'socio-temporal order' against which social action is conducted and measured. Similarly, sociological analysis of time has been concerned mainly with the standardization of these parameters as an identifying feature of modern complex societies.[2]

There have also been critiques that social science has rather uncritically incorporated this orderly time into their models, treating it as "a background or hidden dimension" (Dubinskas, 1988a). Hassard even declared that "[t]ime is *the* missing variable in modern sociological analysis" (1990, p. 1, emphasis in original). Instead of the "linear-quantitative" approach dominant in functional or structural perspectives, Hassard requested more "cyclic-qualitative research" on time that puts emphasis on "creating temporal meanings rather than responding to temporal structures" (1996, p. 586).[3] Bash (2000) also argues that most analysis of time is still "temporocentric" and advices researchers to adopt the social constructionist approach that focuses on the relative, contextual, organic, and socially constructed aspects of time rather than the objective, linear, unitary, and dominant order of time.

---

[2] Many classical works in industrial sociology provide ample examples of standardized and rigidified time that emerged as a by-product of industrialism. Mumford (1934), for instance, argued how the clock became the key machine of industrial age for measuring, coordinating, and controlling the highly specialized industrial work. Also E. P. Thompson (1967) detailed how workers became subject to an extremely elaborate form of time discipline in large-scale factory manufacturing systems.

[3] As examples of this cyclic-qualitative approach, Hassard suggests Roy's (1960) account of time structuring among factory workers, Ditton's (1979) analysis of bakers' time strategies, Cavendish's (1982) portrayal of time battles on the assembly line, and Clark's (1978; 1985) attempts to link temporal experience with organizational structure.

Regardless of the difference in how time is conceptualized, much of the literature on social time can be characterized by the conceptual dichotomies of time, although many researchers tacitly assume one or the other view rather than explicitly address the issue. This conceptual dichotomy is often expressed in various pairs of terms depending on the focus of a researcher, such as clock and event time,[4] linear and cyclical time,[5] monochronic and polychronic time,[6] chronos and kairos.[7] Although these dualistic sets use slightly different concepts, they may be lumped into a single big pair, clock-linear-monochronic-chronos-quantitative-unitary-objective time on one hand, and event-cyclical-polychronic-kairos-qualitative-heterogeneous-subjective (or constructed) time on the other.

These dichotomous conceptions of time are closely related to the conceptual dualism that has been a widespread analytic device in social science. It gives the analyst the advantage of embracing a range of significant variations observed in any given dimension of social life (Bash, 2000, p. 192). In actual analysis, however, the conceptual dichotomies tend to appear as discontinuous and mutually exclusive either/or

---

[4] Clock time, resonating with an atomic or mechanical view of the world, has been associated with the commodification of time, work discipline, and mechanical rhythms in industrial organization. Event time, in contrast, is conceived as qualitative, heterogeneous, discontinuous, and unequivalent, defined by the participants of the event.

[5] The linear time dominated by irreversibility, succession, and history is contrasted with cyclical time with emphasis on the repetitions both in natural and social phenomena.

[6] The distinction between monochronic and polychronic time was introduced by Hall (1983) to describe two different ways of organizing time in different societies. The M-time (doing one thing at a time) is oriented to tasks, schedules, and procedures, while the P-time (doing many things at once) is oriented to people, human relationships, and family. Hall emphasized, although neither pattern is right but only different, that these two patterns do not easily mix, because they are based on different assumptions and values.

[7] Chronos is the "chronological, serial time of succession...time measured by the chronometer not by purpose," whereas kairos, name after the Greek god of opportunity, refers to "the human and living time of intentions, goals...the time not of measurement but of human activity, of opportunity (Jacques, 1982, pp. 14-15).

formulations, often making it difficult for researchers to analyze the wider complexities between the opposed concepts. As a response, a few scholars have attempted to reconcile various dichotomies of time by converting them into recursive, mutually defining and constituting dualities (Adam, 1995; Clark, 1990; Dubinskas, 1988a; Orlikowski and Yates, 2002).

As Adam (1995) aptly pointed out, to transcend the very framework of conceptual dualism, the focus of analysis should be given to the complexity of everyday life where the tensions between the more or less compatible times are managed and expressed. In other words, we need to conceptualize and analyze the connections between multiple times or their mutual implications, beyond merely declaring the duality of concepts. As a way of exploring the link between multiple times in distributed work, I adopt the notion of "temporal structuring" (Orlikowski and Yates, 2002) that focuses on situated practices of team members through which they produce and reproduce a variety of temporal structures and which in turn shape the temporal rhythm and forms of their ongoing practices:

> People in organizations experience time through the shared temporal structures they enact recurrently in their everyday practices...the repeated use of certain temporal structures reproduces and reinforces their legitimacy and influence in organizational life (p. 686).

These legitimized temporal structures serve as a "temporal map" (Zerubavel, 1981, p. 14) that informs and directs the timing and rhythms of members' social action. At the same time, it is through such practices that actors modify the temporal structures over time implicitly or explicitly.

From this perspective, time is not to be found either in the mind of individuals or in abstract, formal, or static structures. It is rooted in the historical, social, material, and discursive practices of actors through which the temporal pattern and order is constituted, reified, modified, and transformed. A practice-based perspective can show how the recurrent practices of social actors constitute temporal structures that are experienced as "time" in everyday life, and yet, how these practices in turn are shaped by previously established temporal structures that influence temporal expectations of actors in organizations.

By looking at the process of temporal structuring, we can access the actual practices through which multiple temporal structures interpenetrate and affect each other's quality and meaning. As I argued earlier, it is not enough to merely recognize the coexistence of multiple times in social life or what Nowotny (1992, p. 424) has termed 'pluritemporalism'—"the existence of a plurality of different modes of social time(s) which may exist side by side." A focus on the ongoing constitution of multiple temporal structures in people's everyday practices enables us to see how different temporal norms and expectations are implicated in any single phenomenon and event that they engage with. For example, a home worker can experience time as a succession of events that are not strictly measured by clock time. Nevertheless, it merely intermeshes with but is often evaluated through the economic notion of time when the home worker reports her hours to get paid or she worries about her productivity with a task.

The focus on temporal structuring in actual practices also contributes to the existing discussion on temporal characteristics of virtual, distributed, or telework, in particular, that of temporal flexibility. It can show how those dominant "ideas" of

flexibility affect the perceptions and practices of time in a virtual team as well as how such "flexibility" is constituted and achieved by local practices through which members attempt to intermesh interdependent but often conflicting temporal structures. Adam (1990, p. 42) notes that "time is fundamentally embedded in the social forms of life which constitute it and which are simultaneously constituted by it." Emerging forms of organizing work—whether it is called virtual, dispersed, distributed, or networked—are not exceptions. A focus on temporal structuring brings to the fore these ongoing practices through which members create, maintain, and change their temporal structures over time.

Distributed coordination and communication

All interdependent work entails the need for coordination. Coordination can be defined as the set of tasks and processes by which groups of actors manage interdependencies in carrying out activities in order to perform effectively as a group (Malone and Crowston, 1990). Coordination has been found important in various work teams (Ancona and Caldwell, 1992; Argote, 1982; Gupta et al., 1994; Hutchins, 1990; Perlow, 2000; 2001), but it is known to be particularly challenging in distributed teams due to the difficulty of maintaining awareness across geographical, temporal, and technical boundaries.

To achieve the coordination necessary for any collaborative action, people maintain an ongoing awareness of events in their work environment. Defined as an "*understanding of the activities of others*, which provides a *context for your own activity*" (Dourish and Bellotti, 1992, p. 107, emphasis in original), awareness reduces the effort needed to coordinate tasks and resources by providing a context in which to interpret and

anticipate others' actions (Gutwin et al., 1996). Kraut et al. (2002) further distinguish between awareness of the task and awareness of the team. Task awareness refers to members' understanding of the overall project, including its history, current status, and future directions. Team awareness, on the other hand, refers to members' understanding of both stable and changing attributes of their partners. Developing and maintaining this awareness has been reported to be much more difficult in distributed teams than in collocated ones (Cramton, 2001).

In collocated situations, members can provide and receive up-to-date information about the status of current tasks and each other's capabilities more easily. Feedback about what other people are doing may be immediate and can be obtained passively without explicit communication. Team members, for instance, can participate in hallway conversations about the group's progress (Kraut et al., 2002) or glance over at another person to see what he or she is working on or how a task is being done (Olson et al., 2002). This passive monitoring of others' activity is conducive to coordination and collaboration. For example, Liang et al. (1995) have shown that members of a team who picked up information about other members while training side by side allocated tasks more efficiently. Similarly, Seifert and Hutchins (1992) have demonstrated that team members could evaluate new recruits and correct their errors by overhearing conversations. Moreover, collocated teams share a context that helps them interpret the information they obtain explicitly or implicitly.

In contrast, distributed teams can go long periods without information about their teammates' activities, often entirely relying on mediated communication to figure out what other members are doing. Delays in remote communication often make feedback

from others difficult to obtain (Kraut et al., 1990; Ruhleder and Jordan, 1999). Without timely feedback, members can appear to be acting independently or hiding the need for interdependence (DeSanctis and Monge, 1999). Hidden interdependence can have a negative effect on coordination (Serfaty et al., 1998), trust in other members, and commitment to goals (Sheppard and Sherman, 1998). The difficulty lies not only in obtaining up-to-date information but also in accurately interpreting this information. Cramton's study of distributed teams (2001) showed that the lack of shared context leads to misattributions of others' behavior, resulting in poor coordination and distrust. On the other hand, Weisband (2002) found that teams in which members periodically gathered information about others and revealed information about themselves performed better.

In a distributed situation, members need to actively monitor others' activities to keep informed about the progress of the team and what other members are doing (Gambetta, 1988). Communication is essential for sharing information related to tasks, informing others about work progress (Rasker et al., 2000), and anticipating others' needs or actions to achieve successful outcomes (Sheppard and Sherman, 1998). Some researchers also have provided some evidence that coordination problems can be alleviated by improving communication. For example, coordination meetings helped work teams generate long-term stability (Maznevski and Chudoba, 2000) and using email to schedule activities helped teams in their coordination efforts (Sproull and Kiesler, 1991).

Cramton (2001; 2002) delves further into the issue and argues that effective communication hinges on establishing "mutual knowledge." Mutual knowledge is "knowledge that the communication parties share in common and know they share"

(Cramton, 2001, p. 346). It consists not only of a specific piece of information, but also the awareness that the others know this information. This second requirement is very important, because without it, people would behave based on their own information and interpretation, assuming incorrectly that their communication partner would share the same.

Cramton found geographically dispersed teams were vulnerable to failures in establishing mutual knowledge. In particular, she discovered five serious problems in the way dispersed teams typically exchanged information: 1) failures to communicate and remember information about context, 2) uneven distribution of information, 3) differences in what information is salient to sender and receiver, 4) differences in communication speed and timing, and 5) uncertainty about the meaning of silence. These failures often exacerbated each other. For example, a member of Cramton's virtual teams sent an email asking for an update, but did not get a response because the recipient was on vacation. The failure to communicate this particular context information then led to ambiguity in interpreting silence. Furthermore, in the teams Cramton studied, these failures of mutual knowledge often led to dispositional rather than situational attribution of others' behavior. For instance, the failure to respond to the email was attributed negatively to the person (e.g., the person is unreliable) rather than to the situation (e.g., the email did not get through). The consequences of failures of mutual knowledge also included the blunting of a team's capacity to learn, failure to meet each other's expectations, and damage to interpersonal trust.

The failures of mutual knowledge and their consequences in dispersed teams that Cramton delineated in her study shed light on why distributed team members have a more

difficult time reconciling issues. In their study of nine distributed work groups in a company's software engineering organization, Armstrong and Cole (2002) observed two outstanding problems: misunderstandings in communication and strangely escalated conflicts. Fragmented communication among distant group members often led to misunderstanding based on different assumptions about the tasks and assignments. Messages were interpreted differently in different sites, sometimes fueling ongoing conflicts among different sites. Conflicts among sites were unidentified and unaddressed longer and flared up more suddenly. Consequently, it was more difficult and time consuming to resolve conflicts among different sites. Mutual knowledge problems explain how these phenomena all interrelate: the failures of mutual knowledge led to failures to meet different expectations, and without mutual knowledge, dispositional attribution overshadowed situational attribution, further crippling members' learning about others' situation and context. In a situation like this, problems take a long time to be detected and resolved, resulting in damage to trust in others.

The prior research is consistent in that the awareness of tasks and team is the precondition for a successful coordination and communication is the key in creating, maintaining, and increasing such awareness. This dissertation attempts to deepen the understanding of this connection, with an emphasis on actual practices and adaptive learning of members over time. Coordination and communication are not static team processes but dynamic capabilities that members have to learn and develop through ongoing practices. What needs more exploration is first how communication is related to members' efforts to coordinate their own work with others' in their daily practices, and

second, how members gradually develop, modify, or transform those practices for better coordination over time, adapting to changing tasks and environments.

Temporal coordination

Distributed coordination depends critically on the possibilities for actors to structure their activities across time and space. While limited work on spatial coordination has appeared in the organizations literature (for recent exceptions, see Hinds and Kiesler, 2002; O'Leary and Cummings, 2004), interest in temporal coordination has been growing (Ancona and Chong, 1996; Ancona et al., 2001; Barley, 1988; Bluedorn and Denhardt, 1988; Blyton et al., 1992; Clark, 1985; Gersick, 1988; 1989; 1994; McGrath, 1990; 1991; Orlikowski and Yates, 2002; Perlow, 1997; Staudenmayer et al., 2002; Walther, 2002).

Zerubavel (1981, p. 64-69) presents two basic patterns of temporal coordination in communities, which he labels "temporal symmetry" and "temporal complementarity." The former achieves temporal coordination by synchronizing the activities of different individuals (i.e., when all group members engage in the same activity at the same time), whereas the latter is based on a temporal division of labor where group members participate in one temporal order yet they do so differently from one another (e.g., when members work in shifts, or when group members work in parallel on different pieces of a joint product). While virtual teams may be able to enjoy increased temporal flexibility by developing their use of temporal complementarity (e.g. allocating work globally to accomplish "follow-the-sun" software development), maintaining temporal symmetry

becomes much more difficult, particularly when members are distributed across many time zones (e.g., arranging a video-conference for an important team discussion).

In his work on temporal coordination, McGrath (1990, pp. 36-44) identifies what he sees as three critical temporal problems in any organization: temporal ambiguity, conflicting temporal interests and requirements, and scarcity of temporal resources. He suggests that organizational responses to these temporal problems can be characterized in terms of three types of activities: scheduling, synchronization, and allocation. By implication, virtual teams are expected to respond to these temporal problems as well, in addition to dealing with the opportunities and challenges that arise from their temporal dispersion. Massey et al. (2003) adopted MaGrath's typology of temporal coordination mechanisms in their experimental setting in order to examine their roles in pacing and synchronizing the efforts of global virtual project teams. Their finding suggests that successful enactment of temporal coordination mechanisms was more often than not associated with higher performance, reducing the time needed to convey ideas and manage the process, and increasing the time allowed for critical discussion.

Massey et al.'s (2003) study is also one of the few that directly addresses the issue of temporal coordination in virtual teams. In their study, however, the temporal coordination mechanisms were designed by the researchers and imposed on the virtual teams, instead of emerging over time from the interactions of team members. In research on co-located groups, Okhuysen and Eisenhardt (2002) followed a similar strategy of studying the temporal coordination associated with process interventions devised by the researchers. While this kind of research helps understand the role of externally imposed temporal coordination mechanisms in team interactions, it leaves unanswered the equally

important question of how temporal coordination arises and is developed by team members themselves.

Whether concerning co-located or virtual organizations, increasing attention has been paid to the importance of temporal patterns or rhythms in coordination. For example, Reddy and Dourish (2002), in their ethnographic study of a surgical intensive care unit, found that different workgroups made use of their understanding of the temporal structure of work in order to help coordinate their various activities. Similarly, Maznevski and Chudoba (2000) found that effective virtual teams were distinguished by a strong, repeating temporal pattern. Unlike Reddy and Dourish's co-located workgroups, Maznevski and Chudoba's virtual teams did not have the luxury of well-established institutional temporal rhythms that guided their members' activities. Instead, virtual teams in their study had to establish a basic temporal pattern through periodic, two-day, face-to-face coordination meetings. However, not all virtual teams are able to have face-to-face meetings to discuss coordination issues up front, and the central question of how to achieve effective temporal coordination of distributed work remains.

On the question of how "self-managing" or "self-organizing" virtual teams coordinate their work temporally, the study of dispersed software development teams by Ocker et al. (1995/1996) provides useful insights. They identified two general approaches to temporal coordination within these teams: mechanisms for organizing group communications and mechanisms for sequencing or structuring work and problem-solving activities. While instructive, this distinction does not address the reality that, in practice, communication within virtual teams is—by definition—implicated in the work, and vice versa. Thus what needs further exploration is how virtual team members

"actually" develop coordination "mechanisms," that is, how do members structure their daily communication and work to temporally coordinate their dispersed activities.

Putting it together: Temporal structuring of distributed work through communication

In organizations, the communication and work practices of members are structured temporally. Not only do members' everyday communication and work practices shape their experiences of time, but their communication and work practices are also guided and structured by the shared temporal norms and expectations of members. As Ballard and Seibold (2003, p. 381) pointed out, "communication lies at the nexus of the relationship between time and work." Organizational members communicate to convey temporal information that is central to ordering and coordinating their work activities, and their communication is, in turn, guided and structured by the rhythm and pace of their ongoing work activities. For example, the announcement of a deadline on a team project leads members to reallocate remaining time across various tasks, while the impending deadline may discourage people from engaging in extended interpersonal interactions to meet the deadline.

Given this recursive relationship between time and work, we can learn how organizational members construct and experience temporality by examining their communicative practices that mediate between the two. The focus on communicative practices is especially pertinent to the study of distributed teams because coordinating distributed work is "communication-intensive." People working in a dispersed way typically depend on more frequent and explicit communication about the temporal

aspects of work than those in co-located teams. In co-located contexts, for example, checking in and out of the office signals the start and end of a work day, and members learn what others are doing by looking over their shoulders or from casual chats during coffee breaks. In virtual contexts, making oneself "visible" (e.g., signaling when one is working, or off for a break, or available for a phone conversation) requires additional activities of explicit communication. In distributed work, it is through team members' communication that a shared "temporal map" (Zerubavel, 1981, p. 14) emerges based on which members orient and structure their activities.

Members' perceptions, constructions, and experiences of time are also reflected in their language and patterns of communication. For instance, the content of email messages can tell us how members coordinate their work chronologically (e.g., "I need to finish this by 9am tomorrow"), or by events (e.g., "We will discuss this during our phone meeting"), or how they establish boundaries between work time and non-work time (e.g., "Please don't call too late"), or how they blur these (e.g., "I'll bring my laptop so I can work while the car is being fixed"). Also, members' patterns of communication—when, why, how, how often, and with whom they communicate—reveal much about the temporal organization of work, such as the span, pace, sequence, and timing of work. Consequently, a careful examination of communication practices can identify important work routines and events that shape as well as reflect the temporal structures of the group or organization. Furthermore, because temporality in organizations is neither unitary nor static, it is continually being negotiated by members through their communication. Thus by examining the everyday communication practices of members, we can learn how

members' ongoing micro-interactions shape the dynamic negotiation and renegotiation of their multiple temporal interests and expectations.

Throughout my dissertation, I will be attentive to this recursive relationship between time, communication, and work. Distributed work is an example of how temporal structures, communication strategies, and social and organizational forms interact, interrelate, and influence each other in member's daily practices, thereby creating new modes of working together.

Site and method

Site description: Little Company (LC)

Established in 1996, Little Company (LC) is a small, primarily self-funded, software start-up company, developing a new programming language product, the LC system.[8] The first functional product was developed about one year after the company was founded and the team continued to improve and market the system in the three subsequent years.

The company was originally formed by four computer scientists, and the fifth member joined soon after. Of these five members, three (Keith, Robert, and Dan) worked full time, developing the product and company strategies. The other two (Fred and Martin) worked part time until mid-1999, attending to marketing and administrative matters while doing a small amount of coding.

[8] Names of the company, product, technology, and members have been disguised for reasons of confidentiality.

The members were dispersed geographically in four different cities and temporally in three different time zones across the US. Dan and Keith lived in the Eastern Time Zone, in two different states but within a day's driving distance. Robert lived in the Mountain Time Zone, and Fred and Martin in the same metropolitan area in the Pacific Time Zone.

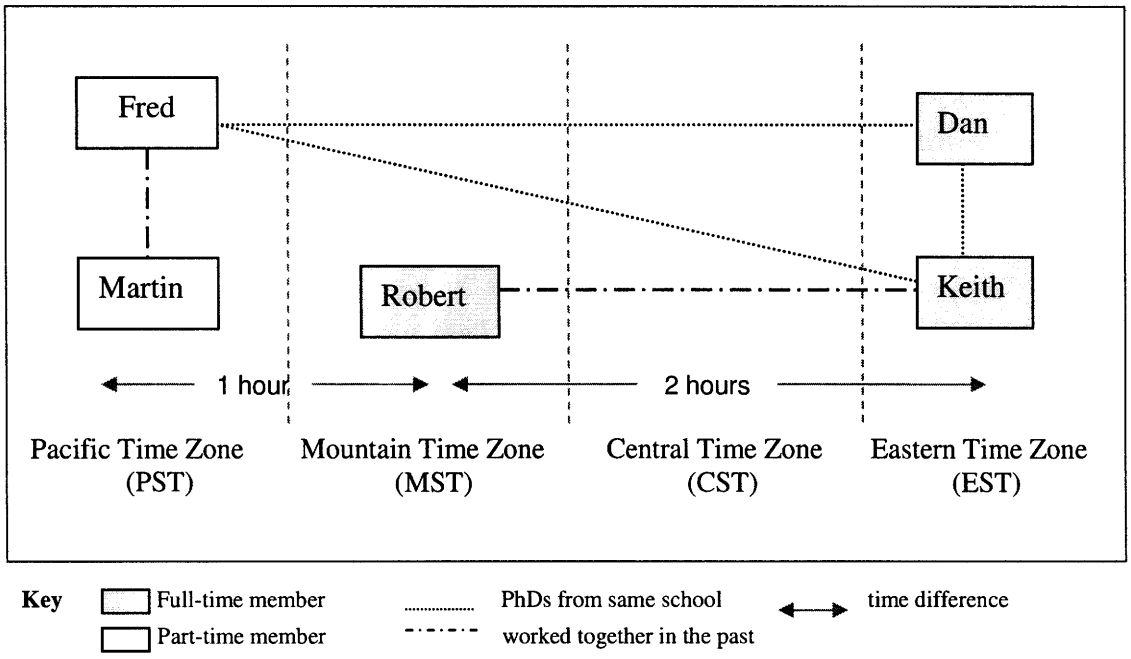
This virtual company had no central physical location, and its central computer server was located at a regional ISP in a town near the home of one of the East Coast members. The three full-time members worked from home and the two part-time members worked out of private offices, using their local computers. Finished and tested components were periodically sent through the Internet to the central server where they were merged together.

Although they were geographically dispersed, there were multiple existing interpersonal ties among the members of this company: three of the members—Dan, Keith, and Fred—had gone to the same college for graduate degrees; Dan and Keith were friends and had written research papers together; Keith and Robert were friends and colleagues at a company prior to joining LC; and Fred and Martin, who had worked together many years ago, were friends who lived in the same city.

Due to geographic dispersion and financial constraint, LC's primary communication media were electronic mail and telephone, with rare face-to-face meetings as a "byproduct" of personal trips. LC never had a team-wide face-to-face meeting during the time the company was in operation. Keith did not meet Martin until after the period covered by my dissertation, and to this day, Dan and Robert have never met Martin.

Figure 1.1 summarizes the geographical and temporal distribution of LC members and pre-existing relationships among them. A more detailed description of the team, its work, and the technologies members used to support their distributed work and communication is provided in chapter two.

Figure 1.1 LC



Data and method

The approach of this research was exploratory and inductive (Struss and Corbin, 1997). As noted earlier, the initial data available was LC's email archive. Semi-structured interviews of LC members were conducted at various points to understand their work, communication, and temporal practices. Later, the data collection and analysis expanded

to include the minutes of their regular conference calls, phone records, and the computerized logs of their code commits.

Altogether, the data I used in my dissertation comes from five sources: LC's email archive, the activity logs of the CVS server, the conference call minutes, phone records, and interviews. Figure 1.2 and Table 1.1 summarize these data sources, and the details are explained below.

Figure 1.2 LC Timeline and data sources

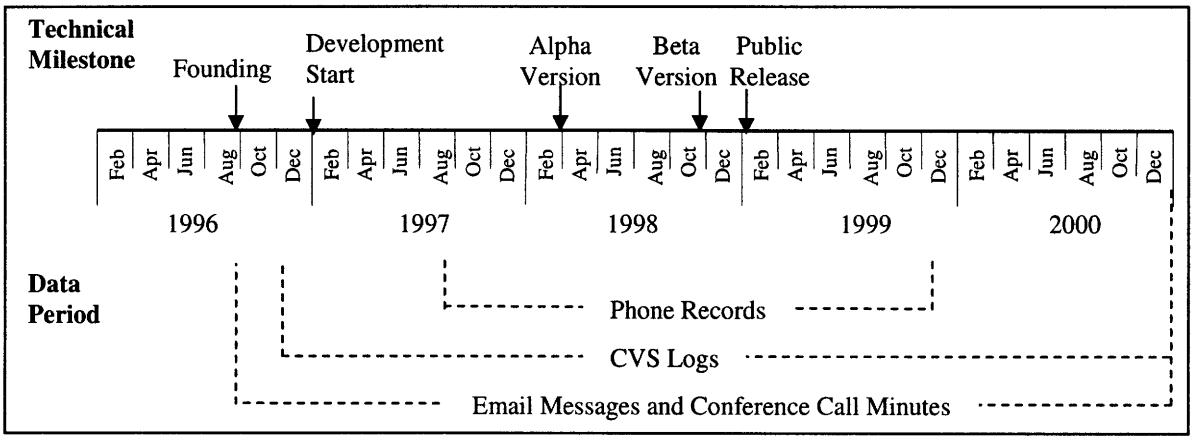


Table 1.1 Data sources

Data Source	Detail
Email messages	11,803 messages (Jul 1996 – Dec 2000)
Phone records	10,149 phone calls (Jul 1997 – Nov 1999)
Conference call minutes	192 entries (Aug 1996 – Dec 2000)
CVS logs	21,522 committed files (Oct 1996 – Dec 2000)
Interviews	7 interviews, average length of 1 hour

Email archive

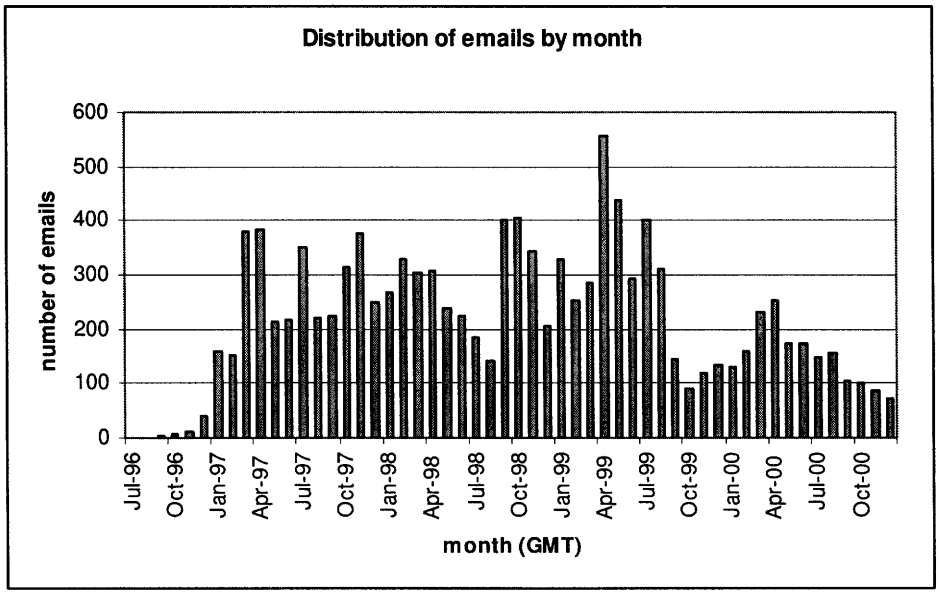
The email archive included all of Dan's received and sent emails, supplemented with messages saved by the other full-time members between July 1996 and December

2000. The final archive (11,803 messages in total) captured most of LC's email activities since the members typically sent their messages to the entire team.[9] The distribution of messages by sender and year is shown in Table 1.2 and monthly distribution of emails over 4 years is shown in Figure 1.3. The final email archive was the main data source for my detailed analysis of LC's activities and communications.

Table 1.2 Distribution of emails by sender and year (GMT)

Sender	Frequency	Percent	Year (GMT)	Frequency	Percent
Dan	3685	31.2	1996	64	.5
Keith	3588	30.4	1997	3234	27.4
Robert	3348	28.4	1998	3352	28.4
Martin	582	4.9	1999	3355	28.4
Fred	346	2.9	2000	1798	15.2
Other	254	2.2	-	-	-
Total	**11803**	**100.0**	**Total**	**11803**	**100.0**

Figure 1.3 Monthly distribution of LC emails



[9] Dan saved all of his emails, including every message sent to the LC list as well as other emails to or from himself. He allowed the research team to use these messages as the base of the archive. The other members were more selective, saving some but not all of their emails. Thus the final archive is missing a few dyadic emails not addressed to or from Dan. Also dyadic emails between Martin and Fred are not included.

Various parts of the LC email archive were coded based on two different coding schemes developed in the course of the research project. Based on a close reading of the email archive, the first coding scheme was developed with categories to identify six dimensions of communication—purpose (why), content (what), form (how), participants (who/m), time (when), and location (where)—and LC's work practices. Two different periods of the LC email archive were coded based on this coding scheme. The second coding scheme, a shorter version of the first one, was developed about a year later, and LC's entire email archive was coded based on this coding scheme. The coding was conducted by four different coders, including myself. Inter-rater reliability tests for the categories in each coding scheme were conducted. All of the tests averaged higher than 0.9 on Cohen's Kappa and all the categories in each coding scheme scored higher than 0.8 on Cohen's Kappa. The coded data was used to identify and retrieve messages of interest, to analyze the patterns of communication by running basic statistical analysis, and also to conduct the genre and genre system analysis in chapter five.

Phone records

LC members used two-way or three-way calling heavily in their daily work. Although we cannot know the content of the phone conversations, the phone bills collected from the three full time members provided access to the basic information about phone communication in LC between July 1997 and November 1999, including the caller, the called, the date/time of the phone call, and the length of the phone call. [10]

[10] The original phone records collected from the full time members covered a longer period (from mid 1996 to Nov 2000), but each member's phone bills covered a different time period. I included only the phone records during the period when the phone bills from all three full-time members were available.

Minutes of regular conference call

LC members also held regular telephone conference calls, monthly in 1996 and weekly beginning in April 1997. During these conference calls, typically running from 45 to 90 minutes, LC members discussed and made important business decisions, coordinated development activities, and assigned tasks to each member. During each meeting, Robert wrote one to two pages of notes, although they were kept for reference and not regularly distributed to the other members. These handwritten "minutes" for 192 meetings held between August 1996 and December 2000 transcribed for research include information about the date/time, people present (or absent), and the key points discussed during the call.

CVS logs

LC members used a central server to store the source code of the LC system. The server also served as a central repository for other company related information. It was owned and maintained by a local ISP (Internet Service Provider) company, and all LC members had access to the server through the Internet to upload (check in) or download (check out) information. In their ongoing software development work, LC members used CVS (Concurrent Version Systems) to keep track of changes made to source code. For each change made (known as a "commit"), a CVS record was generated to include the time, file name, number of lines changed, and author's comments. Following is an example of the CVS logs:

```
revision 1.12
date: 1998/01/15 18:58:48;  author: radar;  state: Exp;  lines: +13 -11
Repairs to choice expression code generation, silenced IO debugging.
```

The original CVS log lacks the information of who committed each file (recorded as "radar" regardless of the actual author as seen in the above example), but by linking the CVS logs and email messages, I identified the person submitting the commit for 68.6% of the total committed files.

Semi-structured interviews

Four of the members of LC were interviewed, omitting only one part-time member (Martin) who had left the firm fairly early in its existence (Nov. 1998). Each full-time member was interviewed individually as well as together in a group interview. Interview topics included the history of the company, team members' backgrounds, technologies they used, their communication, work and temporal practices, their local contexts, etc. Ray, who was hired after the period of this study as CEO to market LC's product, was also interviewed once and provided his observations of LC's work practices from a marketing perspective. Also, the spouse of one of the full-time members was interviewed and provided a valuable account of her experience from the perspective of a home-worker's family. All of the interviews were semi-structured and about an hour long. Except for the last one with a member's spouse, all of the interviews were recorded and transcribed. In addition, one of the full-time members was accessible throughout the research project, serving as a key informant, allowing follow-up inquiries and discussions.

Each data source offered a particular view of LC's communication, work, and temporal practices. The email messages and meeting minutes provide contemporaneous records of communication. The phone records help understand patterns of synchronous communication and coordination. The CVS logs help track LC's technical activities. The

interview transcripts provide information to understand members' work processes and contexts.

The data sources were analyzed in multiple ways. The qualitative analysis included reading and coding email messages, reading interview transcripts and meeting minutes multiple times, doing thematic content analysis, and looking for patterns in the CVS logs and phone records. The quantitative analysis, using basic statistical techniques, was also done to explore and identify patterns in LC's communication.

Structure of the Dissertation

The rest of my dissertation is organized into three main chapters, preceded by a brief chapter that sketches the contexts in which LC operated.

Chapter three examines the perception and practice of time in LC using LC's communication and interview data. I show that the dominant ideas and discourses of new temporality, often represented under the banner of temporal flexibility, were also present in LC, but argue that the "actualization" of the ideas involved the interweaving of heterogeneous elements and practices. In other words, I report how "flexibility" was socially constituted and achieved by local practices.

Chapter four focuses on the work itself, and how it was coordinated in the virtual context using LC's communication and interview data. After examining the actual practices that LC members developed using various modes of coordination and communication, I argue that the way virtual work was coordinated was related to the

temporal experience and norms in LC, in particular, temporal flexibility and autonomy described in chapter three.

Chapter five delves in more detail into LC members' use of email, the main method of their communication, for temporal coordination. Using genre and genre system analysis, I identify and track changes in the form and uses of genres and genre systems developed by LC members. I also examine how these communication genres and genre systems related to various temporal requirements and constraints they attempted to coordinate.

Chapter six is a brief conclusion summarizing the findings of this dissertation and its implication for future research and practices in distributed work.

CHAPTER 2

Putting distributed work in context: Little Company (LC)

This chapter describes the local contexts in which the virtual team of this study operated. Actual practices that LC members enacted to communicate and coordinate their dispersed activities cannot be properly understood separated from these contexts. Three contextual factors are found particularly pertinent in shaping LC's practices: who they were (team), what they did (work), and which tools they used to work as a team over distance (technology).

Team

When LC members started their distributed software development project in late 1996, the idea and practice of distributed teams were still nascent. To their knowledge, other main competitors were involved in similar projects in a collocated way, which, to the eyes of LC members, were neither technologically innovative nor efficiently managed. Keith, the founder of the company who brought the other members into the team, saw the opportunity that he could build a superior product in shorter time-to-market with far fewer people, as the comment below from Keith demonstrates:

> I think that this has been well studied, [and], if time wasn't an issue, three people can really build something much more efficiently than ten people. In other words, there is the nine-men-can't-make-a-baby-in-one-month sort of mentality here, and we work very well together...If we were a 20-person company, most of those 20 people wouldn't be the people developing the core system. But what we did is not going to scale up beyond five people working on the core project, and we have no

> illusion of that. But it's an incredibly effective way. We compare ourselves, you know, I know what the two big competitors to us, which have mediocre free products, I know how many people they have on those projects and we built something which is better than what they built over, roughly, the same period of time with three people, and they had a lot of people.

However, not every group of three people works more efficiently than a group of ten. The initial assessment that he could build a system software product in six to nine months was based on his assumptions about the colleagues with whom he was to work. In other words, the project was perceived to be just the right scope for five people (or for three people on the core project) in a certain timeframe only if he could find the right people capable of doing it. And Keith knew a few candidates who were not only best in this "art" but also would be interested in his project:

> We sort of sat around and said, if we're ever going to make money on what we did our Ph.D.s on, this is it. [LAUGHTER] We weren't just sort of Ph.D.s in computer science, we were Ph.D.s in programming language implementation. We actually had all of the skills to go and do this, OK? And so we were at the top, I mean, we were close, close to if not, [and] we had complimentary sub-skills or micro-skills within the area.

Thus, from the outset, the idea of the technology was infused with the idea of the people who were going to build it. The project became viable because the participants were confident that they possessed the appropriate knowledge and experience for scaling down the complex technological project into sets of "doable problems" (Clarke and Fujimura, 1992) they could efficiently handle:

> I mean, in the sense that we know that we can afford to parcel out the individual tasks just informally between the three of us and just sort of, you do this, you do this, and they'll, and have the pieces drop together. It just doesn't, I mean, I know enough about software engineering, it does not scale out to be bigger than that,

> but for three or even five, it's a perfectly viable way of doing things. And it's an incredibly efficient way of doing things.
>
> When we actually started, we spent a lot of time saying we're going to do this, and laying it out, and we spend a lot of time looking at what we've done along the way and looking at what we need to do and having some perception of what the marketplace needs and what kind of product we wanted to build, and each of us was picking up the slack of, you know, in other words, who's going to do this, who's going to do this, who's going to do this and breaking it down, taking this very large task and breaking it down into a small set of problems that the three of us can handle.

Once Keith found the right people for the project, the project continuously evolved around the idea of who the members were and how they saw themselves. Above all, the very reason why they chose to do a distributed project was to get the right people on board given the financial constraint. The people Keith identified as right for the project happened to be located in four different states, and the distributed project structure was thought to be the solution to serve the organization's need for expertise and members' personal and family needs:

> Most companies say we're going to do it in one place. I think that we got better, you know, I think that we got Dan and Robert and me, pulled together, and we weren't going to get them in one location, period. And so it's a trade-off that you make, at the size company that we are, you know, maybe a little bit bigger, whatever, but you know, you're not going to see a Cisco, you know.

The decision that they did not need to be collocated was largely based on their self image as "highly experienced engineers with a Ph.D," an image that repeatedly surfaced in internal discussions and outside representations of who they were. It was the most important symbol that they used to cast away some misunderstandings that their status of working in a small start-up dispersed across the country might trigger. First, LC members differentiated themselves from other late 1990s *dot com* companies trying to

sell merely an "idea." They emphasized that they were not "just a bunch of fast talkers" or "20-year-olds looking for a huge amount money to do something." Second, especially at the earlier stage of the company history, LC members tried to avoid being seen as individual amateurs excited about a hot technology. Keith wittily said in an interview, "we have gray hair." LC members had more than 30 years of combined experience with the technology, both in research and implementation, and they stressed this fact whenever possible. In his email to Dan, who was about to contact an outside expert to consult on a part of their system, Keith advised Dan to give his academic credentials, adding that "it will also give him the piece of info that you are not some hacker but are fairly skilled in the art, just not this corner of it." Later in 1999 when they had a working product and started to market it on the web, Fred made a similar point on the issue of whether to list the research papers members had published on the company's website:

> Martha is right that you never really see this sort of info in most web-pages. Nonetheless I think we are in a peculiar enough position that these "papers" links are a good idea. They are independent and verifiable evidence that we actually know a lot about what we are doing. Most visitors will not follow those links and most that do will not be much enlightened by doing so. But I don't think it hurt us too much to have them there...and some small subset of visitors will be reassured of our competency by looking at these pages.

Being "highly experienced engineers with a Ph.D." was not only the evidence of their competency and credentials to the outsider. It was also a crucial symbol of a set of capabilities that constituted the team's identity and its competitive advantage. "So there are advantages" over the competitors who "will build out of people who are not well-trained and who don't have a lot of vision," stated Keith in an interview. However, as the flipside of the coin, it also signified what the team was not and what they lacked. They were the "technical organization without marketing savvy," using Keith's expression.

Although LC had two part-time members who were supposed to take care of the business side of the LC project (e.g., developing business models, contacting potential customers, conducting marketing and business relations, managing various administrative issues), it was far from sufficient. Keith precisely pointed it out in his interview while describing how he had brought two part-time members into the team:

> We completely misinterpreted or underestimated how much business help we were going to get out of Fred or Martin. I found everybody knew Fred. Fred had been a friend of Dan's when Dan was out in California and they'd become very close because they knew each other in graduate school. And Fred's particular subspecialty in terms of his technical expertise and Robert's were actually very similar. Fred did exactly the same sort of stuff and that's how Robert knew Fred. So we were very comfortable with that. And Fred found this person, Martin, and Martin was a very good friend of Fred and Martin had more business skills. We really felt, when we started the company that the business side of it was going to be really covered by two people.

Fred had experience in founding and running a start-up, which "Keith didn't see any reason that he needed to learn to do" according to Fred, so he handled most administrative matters around LC (e.g., accounting, tax, legal issues). However, Fred was only relatively superior to the three full-time members in terms of being "marketing savvy," as indicated by his technical background. In fact, he had strong interest in being involved in the technical aspects of the LC project when he joined the company, as Fred recalled in his interview:

> This [LC project] looked like an interesting way to get back into the sort of engineering that I'd been doing five years ago, and maybe have a little bit more involvement in the engineering side, at least in the development side, and a little less involvement in doing business development. I also thought that this new language was probably going to be important in the discipline, and I didn't know enough about it and this looked like a good excuse to learn something about it.

His plan did not pan out. Working only part-time, he had to attend to other administrative and marketing duties and never found time to do much coding. The plan's demise was also due to the limited role of Martin, the other part-time member that Fred recruited into the team to lead LC's marketing. To his dismay, Martin could not provide as much help with marketing as expected and eventually left LC for another company shortly after the public release of LC product. In fact, Martin was "more involved with programming support and tools and less [with] nuts and bolts [of marketing]," according to Fred. Keith, singling it out as the biggest failure of the company that they did not get professional help with marketing in time, expressed similar regret:

> He [Martin] was the one who had the business contacts and who was going to do it [marketing]. The great failure of the company was not realizing that he was not the right person to do this [...] We were inexperienced; we just didn't know what we were doing then, even though we got professional help finally from Ray.

LC did not have professional management or marketing staff during the first four years of their project, and to some extent, they underestimated its importance. For example, shortly after the public release of their product, Dan wrote in an email, "I think we should get the marketing spiel in the pipeline somewhere between now and very soon. I suppose we should hire a marketing person to do this...but I must say it just doesn't look that hard." He continued:

> There's this standard spiel about all the great stuff we do, complete with a bullet-item list that should always be longer than the competition's, plus references to our years of experience in actual research and actual development in everything from supercomputer Fortran to Modula-3 and functional programming to automatic optimizer generation to architecture design. "We wrote the papers that the competition is reading," or something like that.

However, they soon realized that it takes more than their excellent technical finesse to conduct "non-technical" tasks. About four months later, Dan wrote in another email, "I know squat about marketing, except that we need more of it." Since the major development work had ended, various "non-technical" tasks –e.g., developing a pricing model, writing a business plan, or most importantly contacting and negotiating with potential business partners—became the central tasks of their daily work. And they handled them poorly and inefficiently. "It was not our skill," all of the three full-time members admitted openly in a group interview. One member also recalled:

> I remember Keith calling me up and saying, 'I've seen these guys operate. I just cannot do that. I don't have these skills. I can't play the corporate golf game.'

Keith, who was the president of the company as "no one else wanted to do it," acknowledged Fred's testimony, saying "I did not know the terminology."

LC's lack of marketing skills was not only evident to themselves but also discernable to others. For example, relaying comments from someone who reviewed the draft of LC's press release, Dan wrote in an email:

> She is still of the opinion that it needs some change in emphasis and wording. Actually, what she said was, "you can tell that this was written by a committee of engineers and not by a marketing person. You need to keep in mind that the main audience for this is a bunch of semi-literate editors.

LC finally hired Ray as CEO and full-time marketer quite a while after both the part-time members left the team. "He has taken a lot of stuff that we didn't want to do and spent a lot of time not doing," noted Keith. However, having Ray on board did not improve the "marketing spiel" of the three full-time members instantly. For example,

they would challenge each other on a technical point in front of a potential customer. Ray recollected one of these incidents he witnessed:

> We were on the phone with [a company] and there's a very large project that I'm trying to get them to schedule us up. I had Dan and Robert on the phone and I started going off and kind of handing the baton off to Keith and Dan just jumped in. Keith was going down this path, and there was this little trail that was parallel to it but had a little different twist, and Dan took it right down there. So I had a meeting of the minds right after and I said, 'In these phone calls, here's how it works. I'd like to add on to what Keith was saying by going along and not jumping in.' But that's because of the four and a half, five years that they've spent together. They're always used to doing that, where the hand doesn't go up and they blurt it out. That's a lot of the coaching that I have to do. They're all wonderful ideas, but you also want to make sure that they're thinking about what it is that they're saying and making sure that the client, potential client, whoever else is on the other end, understands what's going on before we confuse them.

LC was a team composed of like-minded engineers similar in both their strengths and weaknesses. There was little, if any, hierarchy among team members. Although Keith was the president of the company, he was more or less equal to other members. And this was also the way Keith saw himself:

> I was President because somebody had to be the leader of the company and Dan and Robert didn't want to do it. And I really didn't want to do it either but I had more money into it than anybody else did, and I've kind of been more the cheerleader along the way than the others had.

At most, he was the first among equals, as Fred put it. Asked how the team dealt with disagreements, Robert provided a similar explanation:

> In terms of issues like 'do we get a CEO?,' I decided that since Keith's investment in this company is twice what mine is, I'm just going to let his weight carry and I will say what I think but I'm not going to actually try to manipulate things so that the company does it one way or the other. I'm just going to go along with the flow if other people agree with Keith. And if nobody agrees with Keith, then he usually backs down pretty quickly. In terms of technical issues, however,

> I'm a technical person. That's what I often care about. Keith is the chief technical officer of the company and always has been the person who we agreed would be making these sorts of decisions. But I was initially the manager of deadlock, which was changed to the manager of strategy. But what my position was in terms of resolving arguments was that if there were ever a time and if we could ever not make a decision because we were evenly split, then I got to decide.
>
> [Interviewer] OK. On technical stuff.
>
> Well, I mean, anything. And I never really had to exercise that. We never really had something where it was just one side against the other in terms of a company-wide decision.

The indispensable role of each member to the team also contributed to the comradeship and the cohesion of the team. "If any of the three of us left the company, there would be a big problem for the remaining two," noted Dan in his interview. With great commonality in educational and professional backgrounds, pre-existing ties and relationships among members, and the critical role of each member to the project, the atmosphere remained highly congenial through most of the project.

Work

Software development in general is a highly interdependent and uncertain task that requires a great deal of communication and coordination. Communication and coordination breakdowns are one of the most salient problems in software development teams (Curtis et al., 1988). Coordination is particularly important in software development for the complex logical and temporal dependencies among tasks assigned to individual members. LC members' work involved pooled, sequential, and reciprocal interdependencies (Thompson, 1967). Sometimes, individual modules were combined. At

other times, one member's code had to be completed before the other member could start his contribution. In many cases, members worked concurrently on related modules, passing codes back and forth. The email excerpt below illustrates that coordinating these interdependencies was a part of their ordinary work:

> Date: Tue, 11 Mar 1997 20:14:11 -0500 (EST)
> From: Keith
> To: Dan
> CC: all
> Subject: the assembler
>
> Dan,
>
> I have gotten the assembly part of the assembler finished and debugged. It assembles the weird boundary cases as I expected it would.
>
> There are three tasks on my list:
>
> 1) get the back end of the assembler written and working. This is about a 1 day job that really consists of writing the output files. After that I have to check the opcodes by loading a file with all of the instructions in a debugger and see if the encoding matches what i think I am giving it. This is going to be a bad day of playing cowboys and endians.
>
> I am currently waiting for fred to send me a sort routine for Ocaml and to help with the debugger.
>
> 2) encoding the floating point part of the machine description for the pc. This should be about a day or two. This is well defined unless you change your mind on the set of instructions you want encoded.
>
> 3) starting the pass of running thru the .scm files and generating the assembly code from the data and function declarations of the classes. This is also about a day task depending on how well defined the structures are at this point. If things are not well defined, this could drag.
>
> which one of these tasks needs to get done first to keep you from getting blocked?
>
> also, I put the latest version of the assembler on the server. You must also get the latest version of the generator.
>
> keith

A coordination failure may produce redundancies (multiple people can unknowingly work on the same task), delays (one person can be idle waiting for someone else to complete a prerequisite task) or reworking (a completed task has to be redone in the light of subsequent information) (Whittaker and Schwarz, 1999). LC members were well aware of these problems and tried their best to avoid them. In the interview, Dan explained:

> Suppose right now someone is working on part of his [module] and sometimes he's working on that part of what's "my" [module]. But I know what he's working on and it interfaces with these other things. So I won't mess where he is, and if I can help it, I'm going to avoid that stuff that he interfaces with. If I work on that and he's working on that, then we're both going to be modifying the same files and we will have to worry about not knowing who introduced a bug. Or I could just plain delay him. I could do something that would force him to rework all of his changes. So even though I didn't create a bug, so to speak, the concepts will be such that he'll have to redo all of his work and so his time will have been wasted. And so you don't want to do that.

Nevertheless, LC members were not exempt from coordination failures, as the following email exchanges evidence. Keith added some new features to the system that were prerequisites to Dan's task. However, this change generated confusion to Robert whose code had been working fine with the previous version:

> Date: Thu, 10 Jul 1997 14:28:23-0600
> From: Robert
> To: Keith
> CC: LC
> Subject: Re: just added the stuff to the server
>
> Keith wrote:
>
> > This removes a couple of bugs I had in an earlier version I put on the
> > server. This version does not seem to crash on every fourth or fifth file.
>
> I was able to do a "make clean all" on the version that was
> there this morning. Unless a crash is pretty sublime, I did
> not crash once. Service pack 3?

> Does this new version fix bugs in the compiler, the compiled code, or both? Is the current make setup smart enough to recompile code if and only if the part of the compiler that generates code has changed?
>
> --
> robert

Keith responded with an email specifying how Robert's part could be affected differently depending on the version of code Robert had downloaded:

> I added some features for dan that he needed to work and the first cut of what I put on the server had a couple of bugs in it.
>
> Depending on what time you down loaded, you will either see the pre buggy stuff, the buggy stuff or the post buggy stuff.
>
> As far as you are concerned, if you got the pre or the post buggy stuff, you are ok. There is nothing I added that will effect anyone except what dan has to start implementing. Once he gets that stuff working, you will have to get everything and recompile from scratch. I guess I put the buggy stuff up around 2:00 edt and the fixed stuff at 4:30.
>
> This stuff was very buggy, you would not have been able to compile anything with it without it crashing.
>
> Sorry for the confusion.
>
> keith

But Keith's clarification did not eliminate Robert's problem because the updated system forced his code to be recompiled unnecessarily. In other words, the dependency existed only between Keith's and Dan's code, therefore the change Keith had made would not affect Robert's code. Nevertheless, Robert had to recompile everything because there was no procedure yet in place that could allow him to know beforehand if the recompile was necessary. This problem of unnecessary or redundant reworking led members to stop what they were doing to discuss how to avoid this problem in the future. Keith first suggested limiting the number of system updates to once a day at a specific time or even to only when the change would affect everyone:

> This is a very difficult problem. It is in general impossible to tell if a change in [a code file] is going to affect the code that is fed to the [code file]. The make files are thus quite conservative and will recompile the library if the [code file] changes.
>
> Dan and I have struggled over how to handle this and if you have a better idea, please come forth with it. The changes that I made today really do not affect the output code, they are just there to let dan take the next step. However, there is no compilation system imaginable that could know this.
>
> I think that as time goes on, we (dan and myself) will try not to put a new [version] up more than once a day (unless we screw up) and we will try to time it so that it is put up late on the east coast so that those further west can just start a make before they quit for the day.
>
> Furthermore, we can try to operate in such a way that we do not put anything on the server until it is ready for everyone else. However, this is somewhat difficult since dan is in [a city on the east coast] and I am in [another city on the east coast] and the only sane way to share what we are doing is to use the same cvs tools that we share with every one else.
>
> I am at a loss at this point as to what should be the proper working procedures. Clearly having the stuff on the server broken for several days because two people needed to cooperate on a complex change is a bad way to work. I do not know if cvs has some way that two (or more people can create a private playpen without the others sharing the code playing.
>
> If you have any ideas, please come forward. The really bad alternative is to serialize (i.e. wait for the system to stabilize before working on the library) and then we never get anything out the door.
>
> keith

As Keith himself pointed out, however, his solution for reworking problems would introduce another problem, a serious delay in the overall work flow, if not jeopardize the collaboration between any subset of the members. At issue was how to keep a working version of the system available to the rest of the company (in this case, Robert) while a subset of members (in this case, Dan and Keith) experiment with it, without causing reworking or delay to either party. After some deliberation, they finally came up with a solution, which was leaving a note in a specified text file when an update affected everyone's code and thus required a recompile.

There were many tasks where each member's work joined up together, requiring a great deal of coordination. For example, while Robert was in charge of coding a module

called "Linker," this piece was so interrelated with other pieces that any change to it inevitably involved almost all LC members. Yates et al. (2003) analyzed LC's email threading activities during the first one-year period, and email exchanges on this module constituted the longest "recurrent conversation," which extended over almost the entire year and included 294 messages. This extended exchange illustrates that the need for coordinating complex dependencies among modules assigned to each member persisted throughout the development phase.

Standard software development techniques such as modularization, integration plans, interface specification, and documentation were also used in LC to manage interdependency. However, software development work is somewhat unpredictable and hard to plan ahead. There always remain many decisions that cannot be made ahead of time, unexpected problems, details to be filled in, and mistakes to be corrected (Herbsleb and Grinter, 1999). Fixing bugs or errors is a good example. Also it was not rare for LC members to realize that some particular piece of code was required only after they were already half-way through their work. In addition, members would find drafts awaiting their comments, questions requesting their answers, or suggestions asking for their feedback. As a result, it was typical that the initial schedules were always readjusted to accommodate various kinds of unexpected incidents emerging from their work. Keith noted in his interview:

> Occasionally I would sit there and I would talk things out with Robert and Dan and say, 'Well, I'd like to do this optimization or that optimization.' We'd have a little phone call and chat about it for 10 or 15 minutes. But then, you know, in the end I would just go off and do what I wanted to, and Robert and Dan had their own little pieces that they were doing and they would occasionally chat, and then go off and do what they wanted to do. In other words, I would just do it, make sure that it worked, and check it in.

The recalibration of task schedules occasioned by various contingencies provides some indications why failure to meet deadlines is so prevalent in the software industry. LC was no exception. Their software product, very original and market pioneering at that time, made it hard for members to come up with a "realistic" schedule even with their rich experiences. The result was an overly optimistic estimate in the initial planning. LC members unequivocally acknowledged that they underestimated "how big it was." Robert was quick to point out "everything has taken us longer than we originally predicted," but added, "which is a standard phenomenon [in software industry] I believe." Dan, who was less aggressive than the other two members, also recalled in his interview:

> The initial [estimation of] how long this will take us was wildly optimistic. I knew it was optimistic. I had no idea. Oh, they [Keith and Robert] were talking like 'we'll have a prototype we can show the people in six to nine months'...and I thought that was really aggressive. Well, initially, we were also thinking it would be done a little faster...The dog was up and walking on its feet, but not well, after a year. But if you look at these kinds of projects, they have teams of five to twenty people and two years to five. We were absolutely nuts. Three guys were going to sit down and write a system from scratch!

The prototype they expected to deliver in six to nine months finally came after a year. The biggest source of this discrepancy was their underestimation of time to be spent for "non-technical" tasks, such as marketing, let alone time for realigning the features of their product with the changing market environment and opportunities. In this context, the constant re-planning and readjustment, both in the overall project and in members' daily work, became the norm rather than an exception.[11]

[11] LC released its product for public in January 1999 and made a few small-scale sales. But they found they were unable to successfully market the product, especially in the difficult market conditions after the dot com bust. The development team finally disbanded in late 2001, but one member of the team, Keith, does technology consulting under the aegis of LC. Members continue to be good friends.

Technology

The technologies LC members used to conduct their distributed software development included not only advanced information and communication technologies (e.g., server, personal computers, 56K modems and digital subscriber line, email, Internet, related software) but also more rudimentary artifacts (e.g., pens and paper, timepieces, telephone). However, some of these technologies were more crucial than others for their technologically mediated collaboration— server technologies, Internet, and the modems/DSL. LC members referred affectionately to this assemblage of technologies as their "collaborative technology", although it is not one of those applications specialized for electronic collaborations (e.g., groupware). Keith explained in an interview:

> Well, it's not collaborative technology like Lotus or anything like that. It's collaborative technology in the sense that there's a source code system that you can check your stuff into and merge your changes with other people's changes and stuff like that. So we could actually maintain a code database without having to email code back and forth to each other, and that was really good... the modem[s] that we had were up to speed for doing that kind of stuff...[Otherwise] we really wouldn't have been able to pull the things together, and that technology was just right off the shelf and just worked.

The "collaborative technologies of LC" can be seen as what Clarke and Fujimura (1992) called "the right tools for the job." They were co-constructed with the concept of distributed company. LC members saw in some technologies "out there" the potential that could leverage their "little bitty company" to a reality, as Keith put it in the interview:

> We were going to work at home...We really kind of started this on the cutting edge of when the technology was there for a little bitty company, using modems...and the telephone rates were cheap and you could get unlimited access to the Internet over 56K modems [that] were just barely coming in when we

started, and that was enough. There was some collaborative software in terms of doing a project that worked over the Internet.

The criteria for what tools were right for their job were defined in context given the task requirements and financial constraints. This practical approach is also observed in terms of the "speed" of technologies, one of the most emphasized temporal standards in modern organizations in terms of competitive advantages. For example, the 56K modem, which LC members used during most of the development period, was considered just fast enough for the tasks—e.g., uploading/downloading codes and using emails.

Of course, LC members looked for better technological options and upgraded their technological infrastructure over time. However, it should be equally stressed that the work process was also designed and adjusted to improve the efficiency of a given technology. For example, it was a common practice in LC that members uploaded large code files or ran long tests overnight ("fire and forget") to avoid bandwidth problems and save idle waiting time:

> Subject: fix worked, am testing a full system now
>
> and going to bed, and I'll commit in the morning if things look good.
>
> dan

> Subject: re: new load of stuff on the server
>
> I will download this tonight and FireAndForget before I go to bed.
> My system is pretty useless while this is recompiling.
>
> keith

It was this mutual construction and adjustment of technological tools and tasks in practice that made their technological tools "right" for their job.

Phone and email were their main communication media. Phone was the only synchronous communication method except for face-to-face communication, which occurred very rarely in LC. Multi-party phone meetings require deliberate coordination (e.g., time determination, rules of turn taking), and the rather unconventional way LC members used the phone for a conference call added to the coordination. Instead of the phone conference service provided by the phone company, they used chained three-way calling for five-person phone meetings. Robert explained the technical and financial reasons behind this practice in an interview:

> We learned very early that if we wanted to have conference calls, the only cheap solution was three-way calling. And so, we just simply form a chain of people on three-way calling, which works reasonably well once you get the hang of it. Well, the first couple times we did it, we were all trying to figure out this three-way calling stuff and it was more complicated than it should have been. But you can do about five people with very little trouble. The audio is reasonably good if people can talk on the loud side. We tried initially just to use multiple telephone lines and use conference calling on two-line phones, and that doesn't work at all because you lose too much audio in a connection like that. But anyway, we all got the three-way calling and we've made use of that...and three-way calling is cheap...it's $3 a telephone line.

And once they "got the hang of it," members used the three-way calling for any occasions that needed group interactions. Robert commented on it as well:

> When I am talking Dan and we're talking about "did Keith do this or what does Keith think about this?" one of us will say, "Well, let's try to call him" and we'll just use the three-way calling and pick up the third party that we're talking about. We probably wouldn't have learned how to do that without the conference call thing, but that's an important part of making this work.

Email, asynchronous text-based communication, is neither simultaneous nor necessarily sequential, making the conveyance of contextual information more difficult.

But the asynchrony of email was also what made it a right tool for LC. Galegher and Kraut (1994) observed that teams using email for communication tend to work more independently. Email was appropriate for LC members, who spent much time working independently on tasks such as writing and testing code. Email "worked well after all," as it accommodated "real enough time" (Bennett and Weill, 1997), the meaning of which was defined and negotiated by the members. It was also closely related to the individual members' media preferences. Robert commented on this aspect during an interview:

> Keith wants to pick up the phone and talk to you. He leaves telephone messages that say call me. I would prefer that I get an email message saying call me about such-and-such or here's the problem, three sentences, call me and let's figure out what to do with this. And that doesn't happen. It still doesn't happen. If I want to ask Dan a question....I may call and leave a telephone message if it's something that I really can't write down very well. But more than often, I will send an email message if I have a problem where I have like a stack trace on my screen of something that didn't work and died, I will put that into email and I will send it out, usually to everybody in the company. And with Keith then, I will call him up and say Keith, I just sent out a stack trace of a problem, you should take a look at it and tell me where you think the problem is.

Whereas Robert tried to make email "real enough" by incorporating more information into the email and thus reducing the need for "real time" phone conversation, Keith obviously did not. He preferred "real time" phone calls for the instant feedback.

The ways LC members used the phone and email provide a good example of "technologies-in-practice," the "specific structure routinely enacted as we use the specific machine, technique, appliance, device, or gadget in recurrent ways in our everyday situated activities" (Orlikowski et al., 2000, p. 408). The capabilities of a given technology, which are often presumed to be inherent in the technology, are constrained, enhanced, or worked around as it becomes embedded in actual practices.

One of the often ignored aspects of distributed work is the sheer amount of time and effort that members invest in the configuration, maintenance, and repair of technologies. It starts from setting up the common infrastructure and related protocols of use necessary for people in remote locations to collaborate. Compatible infrastructure was important because a problem in code sharing can be caused by a little disparity, even by different versions of software as seen in the following example:

> Subject: RTF difference
>
> I spent some time investigating the difference between what my MS Word produces and what Keith's does. There are differences in the internal RTF language that cause hundred or so mismatches in Unsafe.rtf.
>
> I suspect the reason is that I use MS Word that comes out of Office 97 and Keith has MS Office 95 and MS "improved" their handling of RTF files between releases.
>
> The bottom line:
> - we either all should have exactly the same versions on the same hardware
> - or we need to "lock" documents (i.e., non-text files)
>
> I'll follow the locking since I cannot at this point "downgrade" my installation.
>
> dan

> Subject: Re: RTF differences
>
> Is Office 97 worth the $200 upgrade fee? If so, we could all move to the 97 version.
>
> I use Word for documents, but mainly I use Outlook for calendar, contact management and e-mail. It seems to be more flakey and requires more space. At this point I have mixed feeling about the upgrade (i.e., I hesitate recommending it).
>
> In any case, Fred, are you still using Word on Macintosh? If that's the case then we still would have a problem.
>
> Martin

In a similar vein, sharing information about each other's "machine" or computing environment was critical to reproduce the procedure on one's own local computer, a necessary step for understanding and helping problems encountered by other locations.

The role of technologies in structuring communication and work is discovered most bluntly when people encounter technical problems or disruptions. In situations such as a computer crash, server shutdown, or bad network, the routine application of technology is disrupted and so is the work. The following email where Keith urges Robert to fix his machine illustrates it:

> I assume that you are not doing updates because your machine still is broken. I think it is time for you to bite the bullet and fix it since it is quite likely that you will have to start reproducing these bugs on your machine. It is easy to send you a single file like this, but by early next week we are going to have to start linking multiple files together [...] When we hit a problem there it will be too difficult to send you the world.
>
> keith

In another example below, Martin reports that he had been trying to set up his new computer system for the last fifty hours:

> From:Martin
> To: LittleCompany
> Subject: My new computer status
>
> I have Win 95, Win NT working –problems with RAS (Remote Access Service). And still need to install lots of software.
>
> I hate Microsoft. I have been doing various setups for the last 50 hours with very few hours of sleep in between.
>
> Regards,
>
> Martin

Members offered assistance to solve Martin's problem, as illustrated in Dan's email below:

> Here's my RAS script. Robert posted directions for enabling logging, and that can be very helpful.
> [....]
>
> Get sleep. Sleep is good.
>
> dan

The next day, Martin finally managed to complete his system set-up. It took Martin at least four days to make his new computer system work properly and it involved everyone in LC at various points in the process. In LC where each individual had to install, maintain, and repair his technological infrastructure, the amount of this technology work was substantial, although it was not considered as "work" *per se*.

So far, I described the contexts in which LC members conducted their daily work: the distinctive team identity and dynamics, the technical and social organization of software development work, and the technologies members used to support their communication and work. Although I separated these contexts from each other for analytical purposes, they influence and implicate, enable and constrain each other through members' ongoing practices. Some such instances have been shown already: the very decision to make the team distributed stemmed from the capabilities of team members; members' technical background enabled them to pick and choose from existing technologies to configure their own "collaborative technologies"; LC members worked around the bandwidth problem by the overnight "fire and forget." These contexts—developing software in a small team of highly skilled professionals through technological mediation—were closely related to the temporal experience, communication and coordination in LC explored in the subsequent chapters.

CHAPTER 3

Enacting flexibility in practice

Working from home interacting with other members in remote sites through technological mediation changes many things in one's life. None of LC members had prior experience of working entirely from home, and now the three full-time members worked out of home everyday.[12] Suddenly, certain habits and routines members had previously formed in the traditional office environment lost their relevance. Some of them were dismissed with little regret, as illustrated by one of the full-time members during his interview:

> Think about how much time you waste driving...that's a free hour everyday potentially. Eat when I want to. If I play music, play whatever music I want to. You don't have to be dressed. You don't have to worry about some idiot telling you can't go barefoot. We don't ever have to worry that much about day care coverage or kid coverage or anything because, in a pinch, I'm within walking distance of day care and the school.

Now, he could eat at any time he wanted, but also gone were the lunch breaks with team mates to chat and refresh. One of the important dimensions of these habits and routines is temporal. Distributed work touches and modifies the ways members experience, organize, and manage time at work by situating them in a novel context. New practices may emerge to replace the old ones in a distributed situation, which is often assumed to allow greater temporal flexibility for distributed members.

---

[12] For the pictures of LC members' home offices, see appendix C and D.

On the other hand, engineers in general have long been known for the intensity of their work schedules and total devotion to work (Kidder, 1981; Kunda, 1992; Perlow, 1997). For example, the excerpt below about engineers' life from *The Soul of New Machine* (Kidder, 1981) leaves little room for family, hobbies, or other interests:

> By signing up for the project you agree to do whatever was necessary for success. You agreed to forsake, if necessary, family, hobbies and friends—if you had any of these left (and you might not if you had signed up too many times before) (p. 66).

Software engineers have also been reported to work extremely long hours with few boundaries between work and life outside (Perlow, 1998). They were found suffering from "time famine," a feeling of having too much to do but not enough time to do it, which forced them to work longer hours (Perlow, 1999). The intense and all-encompassing work schedules, so prevalent and often perceived as inevitable among software teams in the US, however, are far from universal across cultural boundaries (Perlow, 2001). Instead, Perlow argued that it is the ways work is coordinated that perpetuate the work time standards and norms in each organization.

Interestingly, LC members seem to have reached the same conclusion from their own experience. They considered long hours and total devotion as the consequence of "bad planning and poor coordination," far from the necessary condition for software development work. The excerpt from a member's email sent to the rest of the team epitomizes members' aversion to such an all-encompassing commitment and rigid structure. In response to a merger offer from a major competitor with an employment contract on a project that was clearly indicated to be a real intensive job with lots of long hours, Dan in effect turns it down saying:

> In addition I will not agree to an 18-month employment agreement for anything less than $200,000 per year. I've worked for a company that was spiraling in before, and it was awful, and the only thing that could be more awful would be if I had signed some sort of a brain-dead employment agreement that forced me to stay on.

Above all, avoiding such a work setting was precisely one of the benefits that motivated them to sign up for the LC project. They anticipated and highly valued the flexibility from working in distributed mode.

LC members firmly believed they could establish more flexible temporal norms and structures. Nevertheless, the change is likely to be more complex, not just because flexibility is an elusive concept often meaning different things to different people, but also because they had to come up with useful practices to support it as a team. Did it happen at all? Then how did it happen? Was their idea of flexibility in line with their actual practices or did they experience slippages and contradictions? These are the questions addressed in this chapter.

Flexibility: perception and practice

While use of the term 'flexibility' is ubiquitous, its meaning is not always clear. Researchers have pointed out that defining flexibility is difficult, since flexibility is multi-dimensional (Suarez et al., 1995) and polymorphous (Evans, 1991). Flexibility has different meanings in various contexts colored by the particular situations or problems faced by organizations. I do not attempt to provide a single definition of flexibility. Rather, I use the term broadly as meaning "capability to adapt" and focus on how this multifold concept was interpreted and put into practices in terms of "time" in LC. In other

words, it is temporal flexibility, "the capability to adapt which and how many hours one works in the face of new, different, or changing requirements," that I mostly refer to when I use the term flexibility in this dissertation.

Since the 1970s, many employers have gradually adopted a variety of programs to provide employees with more control over their time. A paradox that has puzzled researchers and policy makers alike is, however, that employees take little advantage of those programs all the while they wish for less work and a greater control over their use of time. This paradox is usually ascribed to the normative pressures of organizations, the kind of rules and policies that reward working long and inflexible hours (Barker, 1993; Kunda, 1992; Perlow, 1997). Some go further to contend that people can escape these pressures and regain temporal control by leaving permanent employment for market-based careers (Arthur, 1994; Kanter, 1989; Pink, 2001). However, even contractors are reported to take little advantage of flexibility they believed they had. Contractors usually worked longer hours than when they were permanently employed and continued to maintain rigid schedules (Evans and Barley, 2004).

To explain why contractors rarely limit their hours or schedule their time more flexibly, the following explanations have been proposed: First, the majority of contractors seek to minimize downtime as they see it as a period without pay rather than a period without work. Second, the practice of billing by the hour leads contractors to equate time with money, thus working longer hours for more money. Third, contractors are typically hired into troubled projects at the last minute and this situation exacerbates their proclivity to work long and inflexible hours. Finally, many contractors put in long, even unbillable hours to ensure solid references and referrals for future contracts. Evans and

Barley conclude that contractors revealed a disjuncture between perceived and realized flexibility similar to that found among employees in organizations with flexibility programs (Gareis and Barnett, 2002; Tausig and Fenwick, 2001).

LC members occupy a distinct position, differentiated from both permanent employees in collocated organizations and free agents, that offers an opportunity for an interesting comparison. Should a similar disparity exist in LC, where members univocally claimed flexibility and control over their time as the core of their temporal norms and standards? To answer this question, it is necessary to distinguish between perceptions and actual practices of time and pay close attention to how LC members experienced, interpreted, and allocated time on a daily basis.

Nevertheless, a couple of scenarios can be delineated in advance. First, LC was above all a self-managing team where the bureaucratic control and normative pressure of organization was almost absent. LC members were largely free from the hierarchically based social relations that might force software engineers elsewhere to work long and inflexible hours. For example, Perlow (1999) documented that the cultural norms and accompanying reward structures of the organizations encouraged engineers to comply and be reluctant to take advantage of flexible programs. Cultural norms define how individuals are expected to respond in different circumstances. If LC had a set of different cultural norms, free of pressure for long and rigid work schedules, then LC member might have actually taken better advantage of temporal flexibility with higher control over their work time.

Second, in contrast to free agents whose relation with the hiring organization is only temporary, LC members identified themselves with the company and envisioned

their future career as related to the outcome of this project. From this identification, two different conjectures can be made. On the one hand, with one of them as the founder and president of the company, LC members were not exposed to as much market-based pressure as the contractors, so they might hesitate to voluntarily limit flexibility. Although time was still money for LC members as the window of opportunity to sell their technology was rapidly closing, the simple fact that they were not "billed by the hour" and that they did not have employers to impress may have made the equation less conspicuous. On the other hand, because their engagement with the project was not partial in terms of time commitment, scope of work, and responsibility as was often the case with contractors, LC members may have ended up working longer hours to move the project forward than they initially expected. Taking all these into account, it is possible that LC members may have enjoyed the flexibility of choosing *when* to work but limited flexibility of *how many hours* a week to work.

How many and which hours did they work?

Software engineers in general are known for their fast-paced, high-pressured, and crisis-filled work schedules, with extremely long seventy- or eighty-hour weeks. It is difficult to know how many hours per week LC members actually spent on work. The question was actually brought up in the interviews, but all of the LC members had a hard time answering. "I never, I haven't really thought of it in those kind of terms," confided

Kieth, for example.[13] Requested to give an approximation, however, all full time members estimated independently that they might have worked "at least" 50 hours a week. This exceeds the baseline 40 hour-week that is generally taken as standard for a full time job. When compared to a few statistics available, LC members' estimate is longer than the average of their counterparts, both permanent employees and contractors.[14]

Even if we take LC members' estimate at face value, it is still not all that surprising, considering their status in the professional hierarchy. Each member's position can be equated to the senior professional level or higher—the level that are reported to have the long and often unpredictable hours of work. Early studies have already found evidence for all-encompassing schedules and blurring boundary between work and life outside of work in those higher in hierarchy (Kanter, 1977; Whyte, 1956), and the long and unpredictable work hours are increasingly prevalent in many professional occupations, especially in those under the banner of knowledge work (Perlow, 1998).

However, I would like to point out that "a fifty-hour-week" has more significance as "symbol" than as measurement. When work is open-ended, creative, and highly demanding, it is hard to standardize or fully plan out the work schedule in advance. In these situations, the quality of work output tends to be regarded as more important than

---

[13] Steward (2000) also reported similar responses among the teleworkers she interviewed. They did not calculate time spent on work, especially enjoyable and productive work, and so were unaware of the clock time they actually gave to work.

[14] Hecker (1998), based on the data for full-time technicians, computer scientists, and programmers drawn from the U.S. Department of Labor's Current Population Survey, reported that in 1997, only 7 and 3 percent of permanently employed male and female computer scientists respectively worked 55 or more hours weekly. Among the technicians, only 4 and 2 percent of male and female technicians worked as long. Even among permanently employed male programmers, a group known for working long hours, only 5 percent worked 55 hours or more each week. In contrast, Evans and Barely (2004) found that contractors tended to work longer hours than permanent employees. Among the technical contractors they interviewed, 26 percent of the men and 18 percent of women reported working over 55 hours a week.

the number of hours each worker puts in to complete the work, which was also the case in LC. The distinction between "productive" and "less (or even non-) productive" time was much more common in LC than between "work time" and "non-work time." After all, not every hour is the same in terms of "productivity," as Keith explained in his interview:

> I mean, there are times, you know, I mean like in any kind of job, even when I had the sort of a job that you're either productive, you know, you go through those phases of weeks or whatever where you're either productive or not productive or whatever, and then you go through these weeks where you're very very productive, but I put a lot of hours in.

The same three hours quantified in clock time is not "qualitatively" the same, depending on the quality of the output. The ramification of this emphasis on quality is not clear-cut, but an interesting connection has been made by Steward (2000) in her study of teleworkers. She observed that teleworkers measured their performance and time use against an optimal hour of work with high-quality output. This new type of calculation measured against an idealized standard, Steward argued, drove teleworkers to work longer hours in order to compensate for any shortfalls from that standard. Could LC members find themselves in a similar situation? Although conclusive evidence is hard to find, a close examination of their email communication provides some clues.

Some emails voice members' concerns about "productivity" and offers to put extra efforts over the weekend to bridge the gap. Consider the following example:

> I'll think about this a little more; it has taken altogether too much of my time today. I intended to have the register numbering mostly done, and it is not.

In this email, Dan is pulling himself out of a technical discussion with Keith, worrying about the delay of his own work. To this, Keith replied sympathetically:

I understand the feeling.
I am behind on the rest of the assembler.
I will be in and out this weekend so give me a call or send mail if you want to talk.

In a separate email later on the same day, Keith also wrote to Robert:

I, at least am far behind my schedule. I hope to have the assembler running this weekend.

Robert, who was usually the most meticulous among the three, was not an exception in getting behind the schedule occasionally. In the email below, Robert describes vividly the various distractions—the machine change, visitors, even a strangely behaving cat—that had prevented him from being productive:

Subject: status

The past week was not a productive one for me. I'm still trying to get my NT machine back to where it was before the great machine swap. The packing peanuts are all cleaned up and most of the boxes are back in the truck awaiting a trip to UPS. I believe I've run out of new york visitors to take on cross country ski trips. The cat is acting normal again.

I'm still working on the fix that builds a guaranteed-to-be-correct set of .scm files. Hope to have it done in a day or so...and return to working on gc for large objects.

Whereas members' concern about not being productive enough surfaced from time to time in their email communication, they did not talk explicitly about the opposite situation. As long as they were on schedule, the results stood for themselves. However, it is hard to imagine, and even harder to find direct evidence that LC members would take it slow and relax a day or two if they had exceeded their own standards. The impressions from email data and interviews suggest that they probably just moved on to the next task on their list:

The three of us were all competent. We can all work pretty hard. Well, we were always the people who are working on it 100%.

Considering this, LC members might have worked more than they would actually admit. "I think you pretty much just work as hard as you can at a sustainable rate," commented Dan in the interview. Then, the "fifty-hour-week" can be interpreted to stand for that "sustainable rate" at which they thought they could stay both committed and productive for a long period of time:

> I don't know that it was significantly more than 50 hours a week. The other thing is, if you're going to do it, it's one thing to say oh, yeah, I pulled an all-nighter. You're not going to pull all-nighters for four years.

LC members' involvement with the project was long-term. They might have worked seventy or eighty hours per week at hectic times (e.g., when important deadlines, such as public release, approached), but they were all aware that working long hours all the time for the entire period of the project would be neither sustainable nor productive. As Dan succinctly summarized, no one can pull all-nighters for four years. The "fifty-hour-week" estimate is a symbol that reflects the implicit temporal norm and standard that members considered reasonable in terms of productivity and appropriate in terms of commitment, around which they guided and directed their efforts.

Seen in this light, "fifty-hour-week" is an interesting number. It is more than the standard forty-hour-week of full time workers but less than the seventy-or-eighty hour week often found among software engineers (cf. Perlow, 2001). It expresses the way members viewed themselves, their work, and their life: They were committed enough to work as hard as they could, but they were competent enough not to need to work all the time; They were independent enough, free enough from the organizational pressure to work long hours, but responsible enough to voluntarily put in "a lot of hours anyway"; work was one of their central interests, not one rung on a career ladder for which to

sacrifice other interests and responsibilities in their life. The "fifty-hour-week" is the number that epitomizes their identity as professionals and individuals.

On the question of which hours LC members worked, the interviews and communication data indicate that LC had no preset office hours and members worked flexible and fluid hours. An analysis of temporal patterns in LC email communication reveals that all of the three full time members worked outside of the "nine to five" conventional office hours or the "five day week" (Figure 3.1). Over the 54-month period, about half of all messages (45.3%) were sent outside of the 9 am to 5 pm time frame, with 33.2% of emails sent between 5 pm and midnight. The most active block of time included the morning hours of 9 am to 12 pm, and the next early afternoon hours of 12 pm to 3 pm. Obviously, the block of lowest activity was from midnight to 7 am, reflecting biological sleep rhythms. The weekly pattern of email shows similar results (Figure 3.2). Some (14.9%) were sent over the weekend, while the rest were evenly distributed across the five weekdays.

Seen individually, there are noticeable differences among members in their temporal patterns of communication. Among the three full time members, Robert typically started his day first and tended not to stay too late, whereas Keith typically started later but worked until very late. Dan's pattern shows the most visible fluctuations by hour, with peaks in the morning and early afternoon, and a consistent decrease toward evening. He also worked the most during the night time.

Figure 3.1 Daily temporal pattern in email communication (1996-2000)

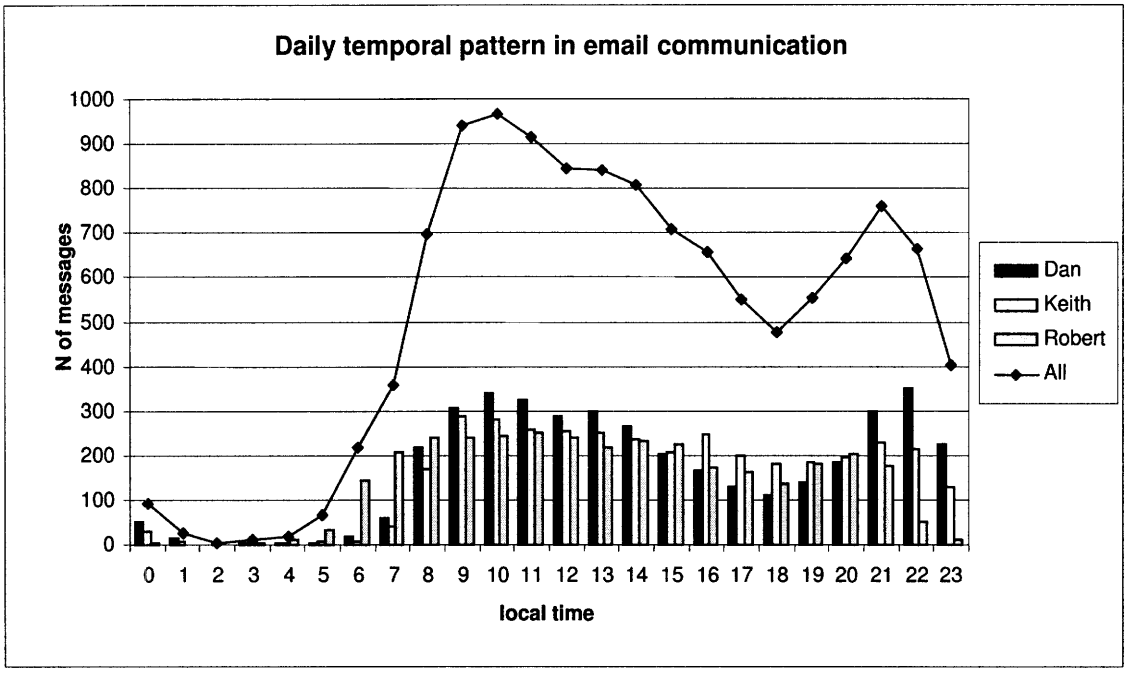
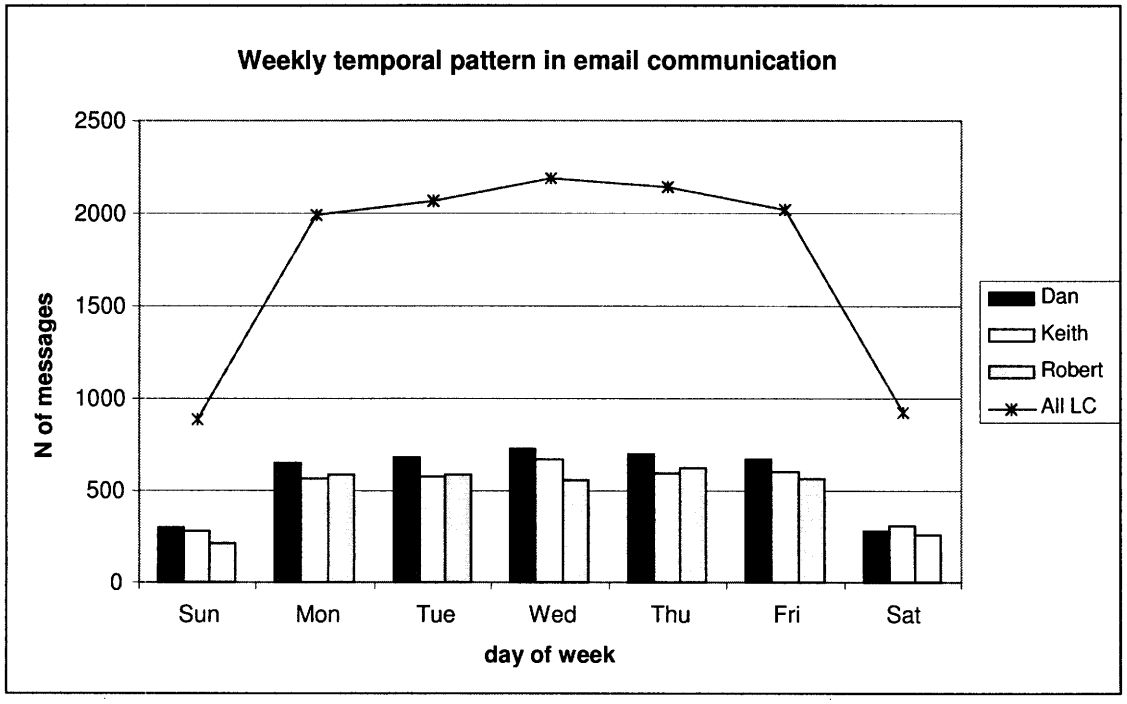


Figure 3.2 Weekly temporal pattern in email communication (1996-2000)



The difference stems from various factors, such as personalities, habits and media preferences, but the most decisive seems to be their different family situations and responsibilities: Dan was the husband of a working wife and father of three young children; Robert had a working wife but no children; Keith was not married but had a girl friend who lived in the same city. Among them, Dan was the one whose daily work schedule was most affected by his regular familial responsibilities: several days a week he had to be a "morning chauffeur" and "drop kids at schools"; he also had to "pick up kids and run them around" in the afternoon, and "get them to bed and read the stories" at night. Dan's day would be different depending on the schedules of other family members, for example, whether it was during the school days or whether it was one of the days of the week when daycare was available, or whether it was the day his wife needed a ride to her office. Hence, he typically started his day pretty early (7 to 7:30 am), but he "often did not start to work until 10 in the morning" and "did not do as much in the evening" (see the hours of 3-8 pm in Figure 3.1). Instead, he had a longer day than the others, staying up late (9 pm to midnight), working safely outside the reach of "three curious children." For Robert and Keith, this kind of "distraction" was little problem, simply because they did not have family members at home during the day to interfere with their work.

The high degree of temporal flexibility in LC is also reflected in members' portrayals of how they structured their day. All three full-time members described their day as "unstructured," or "loosely structured" according to "what they had to do" on a particular day.

In flux Robert said not only his office but also his general work environment was always "in flux":

> Things change as time goes on. But I'm not an orderly person in the sense that...that my desk is rarely clean. And if it's clean, it's only, you know, things start accumulating [LAUGHTER] and stacks of stuff in my office. I like to play around with all sorts of tools and mechanical things...And I'll often just have something sitting in my office that I will play with when I'm thinking about other things. And then, I just set it on the floor and then something else comes in. Pretty soon, the floor is completely covered with things that I've been playing with. As a matter of fact, I start tripping over them and [LAUGHTER] OK, it's time to put things away and. But OK. So, that's kind of my office environment. But my work environment, kind of the first thing I do in the morning is just kind of look at the massive quantities, email, that we get and I have to sort through and just kind of decide is if there is [some]thing that I have to deal with right now. Are there any disasters going on, any new stuff that, you know, somebody found a bug that is something that I should be fixing. And so, I kind of go through that and decide whether there's something there that's going to really change what I thought I was doing today. And then, after I've taken care of that, then it's kind of back to remembering, OK, now what was I going to do today? And I mean, we have telephone meetings that we try to do weekly and so forth. So, if I have something like that on my horizon, I do whatever I need to do to prepare for that. If there's something that I said I was going to do that I should just do and get off the plate, then I should do it before the meeting, I do that. But for the most part, I'm extremely unstructured.

Going by spells For Dan, who had to juggle work and family responsibilities on daily basis, a work day was fragmented by various activities, which, in Dan's words, went by spells:

> I honestly don't know [how many hours/week I worked] because I know my time is broken up. I don't get started right smack-dab at 9 AM. I often don't get started until 10. Some days a week, I can work cleanly until five, but especially during the school year, we've had kid problems...Picking people up and running them around and doing stuff. Then, I would work in the evenings and sometimes I would work part of the day Saturday and part of the day Sunday...I think that it goes by spells.

Working on a walk Keith's description of his work day sounds less hectic than Dan's, interweaving two different kinds of activities—working and taking breaks—at the same time, but he also had high discretion as to how to spend his time:

> I will get up between 8:00 and 9:00 in the morning, and I will essentially start working in the morning, and I'll work, I'll work and I'll take walks, sort of intertwined between the two, because I actually do, I actually know what I'm going, I actually sort of play out in my mind, when I'm taking a walk, what I'm going to do for the next hour, and then the rest of it is just typing it in...Yeah, but I'll take two or three, you know, couple of mile walks a day...if weather [is good], places around my house where I can do that.. So I will just sort of interweave working and, working and walking, and then, you know, and then be at home when I need to be at home, if I need to talk to somebody or whatever. But I'll then work until about 1:00 in the morning, too, you know, and you know, on the other end of that. Now that doesn't mean that I'm really putting in that long of a day because I'll, there'll be breaks and there'll be other stuff that happens in the middle of the day.

LC members had a high level of operational flexibility, to adapt their work schedules daily or even within a day according to changing situations. Once members divided work and agreed upon general deadlines for a list of tasks, members directed their efforts around those guidelines (thus with limited strategic flexibility for individuals to change their commitment to those general task schedules and plans), but they were free to decide how to allocate their time on daily basis to get the tasks done as long as they delivered high-quality code on time.[15]

The following email exchanges among LC members capture how a work day of each member unfolds differently depending on the local exigencies. Dan sent a message

[15] The distinction I made between operational and strategic flexibility derives from applying Bailyn (1985)'s distinction between operational and strategic autonomy to temporal dimension of work. The former refers to freedom to determine the means for achieving the objective, and the latter, freedom to set one's own work agenda.

about the local weather and the problems caused by it, and Robert wittily reacted to it asking whether LC had snow days:

> From: Robert
> To: all
> Subject: Re:fyi
>
> Dan wrote:
> > In case anyone is wondering, we just had a shitpile of snow. Keith's power is out, and my ISP is unusually hard to reach.
>
> Does LC have snow days?
> --
> robert

> To: all
> From: Dan
> Subject: Re:fyi
>
> Robert wrote:
> > does LC have snow days?
>
> Don't think so. It picked a good day to snow; I made a couple of small fixes to instruction-form selection, and have been testing them while I shoveled snow.
>
> For reference, a "shitpile" is somewhere between 18 and 24 inches, packed. A pine tre in the yard next door broke off about 12 feet off the ground, an arborvitae in another neighbor's yard removed itself, roots and all, from the ground. Down the street there's a maple tree with two big branches down, across the front walk of a guy who's about 80. But, lucky us, all our services come in below ground.
>
> Tonight it is supposed to get 20, but tomorrow and the next day it is supposed to be warm.
>
> dan

Without a central office to commute to, snow days lose their conventional meaning, thus conveying a funny twist when mentioned in this tiny distributed company. Still Robert's joke had a point in that the power outage and bad network could be disruptive enough to call for snow days even in a distributed team. Dan shrugged off this concern, saying "it picked a good day to snow," although what really happened was that he picked a good task to do on a snow day. By working on the task that could be done independently,

requiring little interactions with others, he minimized the disruptive effect on his work. The chosen task also allowed time to shovel the snow while tests ran on his computer. If flexibility is defined as the capability to adapt to new, different, or changing requirements, Dan surely demonstrated this capability by working around the problems.

In sum, LC members had the flexibility of choosing when to work or how many hours they worked on a particular day, but the amount of their work load was probably no less than that of other software teams, if not greater, considering that LC was trying to develop a product with the three full time members that would typically take "a team of five to twenty people."[16] In fact, Keith, who was singled out by the other members for working "the hardest of the bunch," admitted that he ended up working more hours after he started the LC project. He also added however, "that doesn't mean that I'm really putting in that long of a day because there'll be breaks and there'll be other stuff that happens in the middle of the day." The indifference of members as to the actual number of work hours is closely related to this daily temporal experience. When different activities are interlaced into a work day, some of which are often done concurrently (e.g., running a test while shoveling snow), conventional calculation of work hours loses relevance since there exists no fine line demarcating time that belongs to work from time that belongs to personal activities. The high level of operation flexibility that allowed LC members to structure their work day in accordance to their local situations seemed to compensate the long day. Going some extra miles caused little concern when members could march to their own beat.

[16] LC members averaged about 100 lines of code per person each day, which is significantly higher than the industry average (Bassman et al., 1994).

Negotiated nature of flexibility

Although all of the LC members valued "flexibility," their assumptions, expectations, and actual practices slightly differed. One way to approach this is examining the different sense of boundary individual members revealed. LC members all experienced changing boundaries between work and life outside of work, as the physical boundary that once separated home and work disappeared. However, each member had a different sense of boundary from the others, stemming from their local situations, personalities, and work styles. For members who structured their work day with high operational flexibility to collaborate effectively, these differences should be addressed, negotiated, and managed. In the absence of formalized temporal structures and hierarchical "boundary control" (Perlow, 1998), [17] LC members had to achieve boundary management as a group.

Embracing flexibility: integrating temporal boundaries

LC members, in one way or another, all embraced the temporal flexibility their new work provided. Their daily temporal structure was not based on a sequential and pre-determined schedule, but on fitting different tasks and activities into a day according to the ongoing situation. In this sense, time in LC was "polychronic" rather than "monochronic" using Hall's (1990) terminology. The following email message from Dan to other LC members illustrates this colorfully:

---

[17] Perlow described various ways in which managers in organizations cajole, encourage, coerce, or otherwise influence the amount of time employees physically spend in the workplace. Boundary control thus refers to managers' ability to affect how employees divide their time between their work and non-work spheres of life.

> Subject: Getting a slow start
>
> I'll be back in time for the conference call, but otherwise I'm not going to get much done this AM. James is still sick, Martha has to get in to work briefly to do time-critical crap there, some paperwork has to be delivered to various people, and it has started to snow. So, rather than put stuff off till the weather gets worse, we're just going to run out and do our errands in a batch. I'll take reading material for when I have to sit in the car.

Robert, describing his "multiple modes of work," gives another typical example of polychronic time, but one that sounds more peaceful:

> I have multiple mode of work. One is that I'm doing some kind of development work that requires, I mean, I'm testing things, building things that take a long time. Some times, these things take hours. I'll go [to do] something else. I'll go mow the lawn. Often I have other stuff that I need to do. You know, dishes are stacking up in the kitchen. I need to do the dishes, and this is the kind of the best time to do them because I've thought about all these things. I am going to be able to come back to it and check out and test all these things.

Dan's email and Robert's comment illustrate that both of them were blurring and integrating multiple temporal boundaries at the same time. Dan worked while sitting in his car and Robert did house chores while his computer was running a test. The same tasks of building and testing code that are often suggested to contribute to the long hours of software engineers at work for the sheer amount of time they take do not pose much constraint to the flexibility, as members utilized the time doing other stuff. It explains why software developers were found to have the most flexibility among various occupational groups of technical contractors (Evans and Barley, 2004, pp. 27-28), quite contrary to the wide-spread perception that software development work requires rigid and all-encompassing schedules.

In general, it was up to LC members how to use their time across different kinds of activities on a particular day. Sometimes, this decision was opportunistic and emergent; as Dan put it, "if you do sort of get too behind on home stuff, then you might take a break

and do that." Sometimes, it was habitual, such as Robert's weekly routine of going to garage sales:

> Friday morning, Saturday morning, I get on my bicycle with a bicycle trailer, ride around town and buy other people' junk. And I come back Friday at 9:30, and ten that's when I start working.

Sometimes, it needed some planning, as Robert describes how he decided to put up a fence one day:

> Part of this was spray painting stuff that you had to do four hours [before rain]...and it rains every afternoon here these months, through the end of June, at about 3:30 or 4 o'clock. And so if you're going to do something like that, well, you get up in the morning, six o'clock, you get prepared, you get the stuff done by 10, and then you go to work.

They did not have to "report" to other members if they left for a few hours to attend to personal issues such as automobile maintenance, a dentist's appointment, a child's school concert, or a garage sale. A simple notice sufficed in most cases. There was no set limit on the number of personal days that individual members might take and members had few restrictions in choosing their vacations, although they would give at least a week of advance notice to coordinate. Even in rare cases where members gave a very short notice or failed to inform others in time, absence from work to accommodate urgent demands outside of work was well tolerated and approved on an ad-hoc basis.

Embracing flexibility involved members' willingness not only to accommodate demands outside of work but also at work. Before deadlines, members worked longer hours together to get the system ready. They also willingly adapted their schedules to help other members. In case they were away or unavailable for any reasons, they would

follow up at their earliest convenience. The following email exchanges provide an example. Robert sent an email to Keith very late, around 1 a.m. in Keith's local time, soliciting some help:

> To: Keith
> Subject: still up?
>
> Are you still up? I'm having problems with crashes in
> ControlFlow.BasicBlock.getSuccCount
> where it expects the BasicBlock field 'target' to
> be non-null. How do I find out what s-experssion is
> causing this BasicBlock to be generated?
>
> robert

It was a Saturday night, and Keith missed the message. But he replied the first thing in the morning, in a message composed entirely of a subject line, "I am around now, if you are there." And Robert did not fail to seize the opportunity:

> Date: Sun, 8 Feb 1998 09:05:39 -0700 (MST)
> To: Keith
> Subject: Re: I am around now, if you are there.
>
> Yes, I am here.
>
> robert

Members also used personal occasions as an opportunity to meet other members or attend business in that area. As a self-funded start-up company, LC had little time and money to afford face-to-face team meetings. Robert explained it in his interview:

> After we actually started the company, we never met [as a whole team]. We talked about all going to some conference at some time so we could at least sit down and talk to each other. We never did that. Part of it was we were always fighting fires and stuff. And also part of it was we never really felt like we had the money to burn to do that.

Rather, members used personal trips to meet other members in the area. "If I was going to be in California anyhow, then I would try and meet with Fred. If we were driving south and passing through [the city where Keith lived], we might stop and meet with Keith," described Dan in his interview. Part-time members, who committed their time to both LC and the other job, blended not only personal and work time but also the time they spent on different jobs. Fred's email below illustrates this very well:

> Date: Fri, 15 Aug 97 18:08:33 PDT
> From: Fred
> To: LC
> Subject: trip to east coast around 11 Oct
>
> Gentlemen:
>
> I will be traveling to the northeast (city A and city B) around the weekend of 11 October 1997. This was originally planned as a mini-vacation for my wife and I to visit friends and relatives in the city A area but I have made plans to visit some [his other company's] customers in city B for a day or two and I will be driving up...to talk to Keith about some questions/problems in some of the LC-generator code that he would like to get cleaned up.
>
> I tell all of you this just in case anyone can think of anything else that I could do on behalf of LC while I'm in the area; e.g. prospective customers to try to contact, consultants to interview, a really cool trade show we should send some one to, ...
>
> I really hope there is nothing else that needs doing, but I do **not** like spending time on airplanes, and so long as I am going to do it anyway it may as well be as productive as possible.
>
> regards..fred

Fred was going to spend a weekend in east coast for a family vacation, but decided to extend his trip to meet customers of his own company as well as Keith who lived in the area nearby. At the same time, he was also soliciting ideas for other possible tasks that could be done for LC during his travel. This example colorfully shows how multiple temporal boundaries interpenetrated each other and integrated into members' daily work and life.

Limiting flexibility: establishing temporal boundaries

While LC members enacted temporal practices that blurred traditional temporal boundaries, they also tried to construct some limits to the flexibility. To be truly flexible, work time should be bounded, not boundless. The same was true with home time. To be productive, members had to reestablish some boundaries between work and home. The ways of creating bounded time varied. Sometimes, members required "quiet time" (Perlow, 1997) by making themselves unavailable for a certain time of a day or by restricting the media through which they could be contacted. For example, on one of those extremely hectic days before an impending product release, Martin sent around an email message before six in the morning (Pacific Time) to ask advice on a technical problem. At the end of the message, however, Martin added a caution not to call him until 7:30 in the morning:

> Date: Thu, 29 Jan 1998 05:54:03 –0800
> To: All
> Subject: what does it mean?
>
> I got it yesterday (source code as of Tue evening):
>
> Possible problems:
> 1. the mapped IO may be reading something wrong out of one of the files
> 2. the mapped IO memory area is messed up by something else
> 3. anything else
>
> Before I try to jump on this, I'd like to ask you to look at the stack trace and help me determine possible cause for the null pointer.
>
> Martin
>
> PS. Please do not call me yet, if you need to talk to me, my family would "kill" me if I awoke them at this time. You may call me after 7:30 AM PST.

Sometimes, members wanted to step away from work completely for a certain period, for example, during a family vacation or a personal trip. However, the way and

degree of (re)establishing boundaries differed by individual members. Robert compared his style to Keith's in the interview when he described his bike vacation:

> I do these [bicycle] trips and I am not the person with the cell phone. And this is one way that I just go out and disappear. And Keith is getting better with this but he has this tendency of saying 'you've got to keep connected, you've got to find a pay phone, you've got to call me.' Keith will plan a vacation and he was out in Tahoe this spring, and it wasn't all that different than when he was in his office at home. He would be sending emails, he would be calling you on the phone. We could call him on the phone, that sort of thing...and that's not my style.

Different from Keith who was always working on the go, Robert did not carry a cell phone, a laptop, or even a watch to enjoy his "leisure time" away from work. Although Robert managed, reluctantly, to make a few phone calls to check in with Keith during the trip, he strongly preferred to be "off-the-hook."

For Dan, the advantages of working from home for juggling work and family responsibilities were evident. However, the integration of home and work did not always go seamlessly. Working from home introduced changes in daily life not only for Dan but also for the rest of family members. The fusion of work and home leads to the interpenetration of multiple temporal structures that were previously more closely coupled either with home or with work. As "work" has been transformed, the meaning of "home" itself has to be redefined. It requires the home worker and his family members to readjust and realign assumptions, expectations, norms and practices related to work and home, which takes time. Dan had similar problems. While Martha, Dan's wife, hinted in the interview at her occasional distress from the change her husband's working from home introduced, saying, "it's just that he was there all the time," Dan also experienced some difficulty in the new situation:

> Sometimes people expect you to do things that they would not expect you to do if you weren't there...oh, just can you do this, can you do that, can you remember to do this kind of stuff.

One strategy often found among employees struggling to balance work and life outside work is to physically separate work and home life, what Nippert-Eng (Nippert-Eng, 1996) called "segmenting." Some home workers have been reported to adopt this strategy, for example, demarcating a part of home as office and sticking to the routines and conventions of the traditional office there (e.g., office attire, office layout). For Dan, who shared his home office in the basement with "three curious kids," physical segmentation was very limited. Although he had a high tolerance for distractions, he found it often hard to concentrate on his work:

> I can work with all sorts of noise going on. [But] I don't deal real well with someone asking me a question right in the middle of trying to do something. So if the kids are just off playing a stupid game, that's OK. If they're asking me questions, then I'm in trouble...You've got to get the kids in the bed. But then they stay up all night and they want books read. I haven't gotten much work done these last few months in the evening.

Instead of the physical boundary that separates home and work, Dan's boundary was temporal: bracketing certain blocks of time each day with fewest distractions from the other family members to work. He "tried to run downstairs about 7 AM and just look at the mail to see if there's anything" before he went off to do his duty of morning chauffeur, and he stayed late for work after kids went to bed.

In addition to the boundaries between work and life outside of work, members also had to maintain a balance between time they could concentrate on their own work and time they could make themselves available for work interactions with others.

Working from home freed members from constant distractions in the office, but it still required daily interactions to coordinate with others. Dan commented on this point, classifying his work into two kinds:

> There're really two kinds. If you're solving a problem all on your own and you're really sort of working and you're not necessarily dealing with anyone...the other thing is debugging and testing...that often requires chit-chat with other people because you'll find a bug and you'll say hey, what about this, did you know that and typically, oh, I think this one's yours.

Members were usually very responsive to the other members' requests for interactions. However, there were also certain periods during which they were busy with a difficult task that required a high degree of concentration and did not want to get sidetracked. On those occasions they needed quiet time. As Dan put it, "sometimes you would say, oh, I just can't."

The influence of interference and interruptions on their own work differed by individual. In LC, Robert seemed to be most affected by such interference, given his work style of keeping things in order:

> I am not a sort of person who can do five things at once. If I'm in the middle of a big project and something else comes in that is deemed to be so much more important that I should start working on it, it's often the death of what I've done. And I'm probably the only one like this in the company, but I cannot concentrate on multiple things, multiple complicated things at once without kind of resolving one and wrapping it up.

This difference is also related to Robert's clear preference of email as communication media, which is less intrusive than phone. In contrast to Keith, who used phone more frequently to discuss technical problems on daily basis, Robert preferred to get most information by email to minimize random interferences:

> Keith wants to pick up the phone and talk to you. He leaves telephone messages that say call me. I would prefer that I get an email message saying call me about such-and-such or here's the problem, three sentences, call me and let's figure out what to do with this.

Negotiating flexibility: flexibility as social accomplishment

As shown so far, it took time for LC members to establish their own temporal structure situated in a new context, which involved both blurring and establishing new boundaries. At the same time, they also had to learn about others' temporal boundaries to effectively collaborate. Individual flexibility was one thing, organizational flexibility was another. Left unaddressed and uncoordinated, the former could hamper the latter. Flexible organization of work as a team required negotiating individual flexibility enacted by members, which involved collective boundary management.

The episode in which Robert succeeded only partially in his attempt to "go out and disappear" illustrates conflicts and negotiations around different temporal boundaries and expectations. Robert recalled the incident in the interview, calling it "a total fiasco":

> Three years ago, I bicycled to Oregon for a family reunion. So, I had this fairly straightforward schedule. I needed two weeks to get there. And I needed two weeks to get back. Well, Martha [Dan's wife] was pregnant at that time and Keith wanted me to do the return trip after the baby was born because Dan was going to be less accessible then, so we could get some stuff done in this period before the baby. Well, I didn't want to do this. But after about a week, in the middle of Wyoming, I called Keith. Keith insisted that I call him every couple days. And I said, 'Keith, it's really hard to call'. And he said, 'well, why can't you just stop at a pay phone'. I said, 'you do not understand Wyoming'...And I am talking to Keith and I say, 'OK, I will change my plans'. He said, 'we'll fly you back.' And I said, 'OK, buy a round trip ticket for me...leaving on such-and-such a day, returning on August 1st.' So, I left my bicycle out there and flew back. Well, it was a total fiasco because Martha had [her baby] early, and it was just totally worthless in terms of what Keith had hoped that would happen. And I ended up coming back in this really hot period of time, in the middle of august.

Although Robert was officially on vacation, he made several compromises to accommodate the team's need. During that time, LC members were busy with writing a big module, a huge task that was added to their development project unexpectedly, adding four times as much code to their system as they had originally planned to write. As it was not included in the original plan, all of three LC members "picked chunks and worked on them," as Dan put it in the interview, which required a close collaboration and coordination among the three. In addition to the increased time pressure this new task added to the development schedule, Dan was expecting a baby soon, so Keith wanted Robert to delay his return bicycle trip until the birth of Dan's baby to maximize time in which all three of them could still work together. Instead of having work constantly flowing back into his vacation, Robert suspended his trip and flew back to assist the team.

The episode around Robert's bicycle trip clearly shows the negotiated nature of flexibility. It also indicates that a single LC member necessarily needed the understanding and cooperation of others to organize his own time more flexibly and effectively. LC members seem to have learned this very quickly, and it is reflected in their communication practices. For example, Robert explained his different approaches to communicate with the other two full time members in consideration of each member's different situation, communication style and media preference:

> If I want to ask Dan a question, I will send him an email because I know there's certain period of times when he's giving kids baths, he doesn't want to talk, he's doing dinner, he's gone to pick up his wife…I may call and leave a telephone message if it's something that I really can't write down very well. But more than often, I will send an email message if I have a problem where I have like a stack trace on my screen of something that didn't work and died, I will put that into email and I will send it out, usually to everybody in the company. And with Keith then, I will call him up and say Keith, I just sent out a stack trace of a problem, you should take a look at it and tell me where you think the problem is.

As suggested in Robert's comment, Dan could maintain and take advantage of the flexibility to juggle his work and family responsibilities, because other members were aware of, respected, and supported his temporal rhythm that was very different from theirs. This clearly demonstrates that temporal flexibility cannot be properly discussed or evaluated by focusing solely on individuals. Only when seen as a social accomplishment, can the ways that people cooperate or negotiate to enact more effective temporal structures, be properly addressed and understood.

Ample examples can be found around coordinating phone meetings in LC. In most cases, members kept the weekly schedule,[18] but there were occasions when a member wanted to hold a meeting to discuss a specific agenda or a member needed to change the scheduled meeting time for personal reasons. As a phone meeting required temporal co-presence of all five members, it was not always easy to find an alternative time. However, an analysis of 264 emails sent to coordinate a total of 83 phone meetings from 1997 to 2000 revealed that these requests were accommodated in most cases. Table 3.1 shows that members succeeded in scheduling 12 additional meetings out of 16 requests (75%) and rescheduling 30 out of 31 meetings (96.8%).

A close reading of emails revealed that it was Dan's family-related schedule conflicts that triggered the most rescheduling. Out of 23 meetings where the source of problem could be identified from the content of emails, Dan's situation accounted for 11 cases, of which members managed to reschedule 10 cases. Juggling responsibilities at home with work, Dan was also the one who requested last-minute schedule adjustments most (Figure 3.3). Among the 83 cases, 18 cases were triggered by an email initiated on

[18] See the section one in chapter four of this dissertation for the detail of how these weekly phone meetings emerged in LC as process intervention in the early phase of LC project.

the same day the meeting was originally scheduled. Among the emails requesting last minute changes, 9 (47%) were from Dan, 6 (32%) were from Keith, followed by Martin (2, 11%) and Fred (1, 5%). It is interesting that Robert never made such last minute requests. This reflects his personal style: he liked things organized and did not like sudden changes.

Table 3.1 Main purpose of coordination through emails regarding phone meetings

Main activities		**# of cases (meeting)**	**Percent**	**cases w/ actual meeting**	**percent**
Scheduling a new meeting		16	19.3	12	**75.0**
Rescheduling a meeting		31	37.3	30	**96.8**
Canceling a meeting		2	2.4	0	
Announcing a meeting		4	4.8	4	**100.0**
Confirming a meeting schedule		14	16.9	10	**71.4**
Others	Notice of absence or limited availability	7	8.4	5	**71.4**
	agenda	4	4.8	4	**100.0**
	Phone logistics	5	6.0	4	**80.0**
Total		**83**	**100.0**	**69**	**83.1**

Figure 3.3 Reasons for last minute adjustments

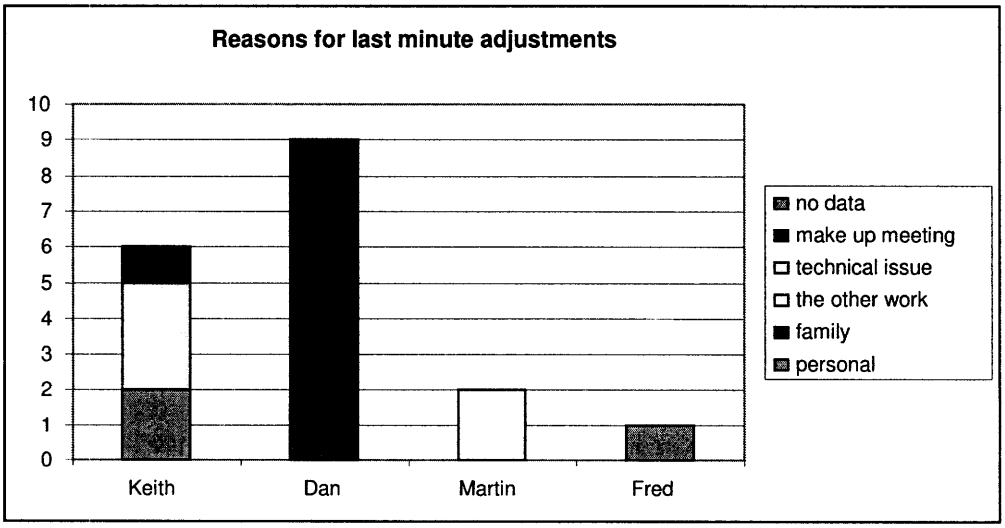


Figure 3.3 also shows the main reasons for which members made last minute requests to change the meeting time. Dan's were all related to his family situation: a child's school event (1 case), his wife's schedule (2 cases), picking up children from school (3 cases), childcare and babysitting responsibilities (4 cases). In more than half of these cases, he was just running a bit late and asked to postpone the meeting by 10 to 30 minutes. Most of the time, the rest of the members willingly waited for him and the meeting went well, except in one case where he came back too late for the meeting. Dan's difficulty in meeting the regular schedule was related to a change in his family that affected his daily life immensely, his daughter's birth. By contrast, Keith's requests were mostly related to work. Consistent with his frequent phone use, he wanted to schedule a meeting to discuss technical issues emerging from their current tasks, instead of waiting until the next regular meeting. For Martin, who worked part-time for LC, the main reason was conflict with his other job. Overall, last minute requests to change meeting schedules were tolerated, and members tried to accommodate other members' situations. In fact, that was precisely how Dan managed to participate in most phone meetings, in spite of all the constraints. He only missed 4 meetings out of 181 phone meetings from 1997 to 2000. Behind this almost perfect attendance record was the other members' accommodation to Dan's effort to bridge multiple commitments.

The negotiated nature of flexibility becomes even more apparent in cases where one person's flexibility becomes a constraint to another. Robert provided examples while describing the advantages and disadvantages of distributed work:

> So, I really do not have, and one of the aspects of a company like this where there's nobody who you're going to meet in your office in the morning who's expecting you to be there so you can talk things out or whatever, I have that

> freedom. On the other hand, when I run into something and I need some information from Keith and I pick up the phone and call him and he's on his walks, or Keith wants to get information from me and he knows I'm out running around to garage sales, it can be kind of frustrating. But we live with that.

Robert's comment revisits a dilemma between control and predictability in a new context. Employees in the modern economy are said to have lost their control over time at work as they moved from the farm to the factory and later to the office (Barley and Kunda, 1992; Edwards, 1979; Owen, 1979; Thompson, 1967). However, this change also introduced the significant increase in predictability that people experienced over the temporal boundary between work and life outside of work. As Zerubaval (1981, p. 166) pointed out, "The very same institutions that are directly responsible for much of the rigidification of our life—namely the schedule and the calendar—can also be seen as being among the foremost liberators of the modern individual!"

If modern employees have exchanged control over their time for predictability in their work lives, various flexible programs can be seen as an attempt to regain the control with some compromise in predictability. When everyone is working fluid and flexible hours, it is difficult to predict others' schedules, hence to coordinate. The high level of operational flexibility in LC also posed similar problems. It was important to find a way to sustain some level of predictability without compromising individual control over time. For example, the attempt to increase predictability by checking up with each other every now and then would cause constant interruptions. It will merely replace the demand of co-presence with that of constant availability. The key was to find a viable way to create a shared "temporal map" without hampering individual flexibility.

Creating a shared temporal map of the group helps members plan and adjust their individual schedules. As Reddy and Dourish (2002) pointed out, information about temporal rhythms in organizations is itself an important resource for workers to help them accomplish their work. However, knowing others' temporal rhythms takes time. In LC, where members enacted multiple temporal structures with high flexibility and autonomy, creating and sharing knowledge and information of others' temporal rhythms was itself an important part of the learning process. In LC, this knowledge was achieved and shared mostly by close "monitoring" of each other over time. In distributed situations, this monitoring is only possible through communication that requires active transmission of information about their status and contexts. Dan described the importance of "checking in" with other members for efficient coordination of work:

> Well, you do check in. It kind of depends. If you know that someone's working on something and they're going to need feedback from you, then you don't want them to be stalled. I mean, just in terms of running the company efficiently. You don't want someone twiddling their thumbs if they would rather be working and they need your help.

Creating and maintaining a collective temporal pattern is another important way to develop the shared temporal map. The weekly phone meeting in LC provides a good example here too. An analysis of the meeting time revealed emerging temporal patterns in LC associated with the weekly phone meeting. Out of 181 meetings held between Jan 1997 and Dec 2000, 71.8% were held on Wednesdays. Figure 3.4 shows that the proportion of "the Wednesday meeting" remained almost constant across all four years, providing an anchor for organizational temporal rhythms. Unlike meeting day, meeting time underwent some changes over the 4-year period. Amongst the 160 meetings for

which time information was available,[19] 42% were held at 3 pm EST/ 1pm MST/ 12pm PST. The second most used time slot was 2:30 am EST /12:30 pm MST /11:30 am PST (22.7%), just 30 minutes earlier than the regular time, with 10 am EST /8 am MST /7:00 am PST (6.1%) as a distant third. 70.7% of the meetings happened at one of the three time slots.

Figure 3.4 Meeting day

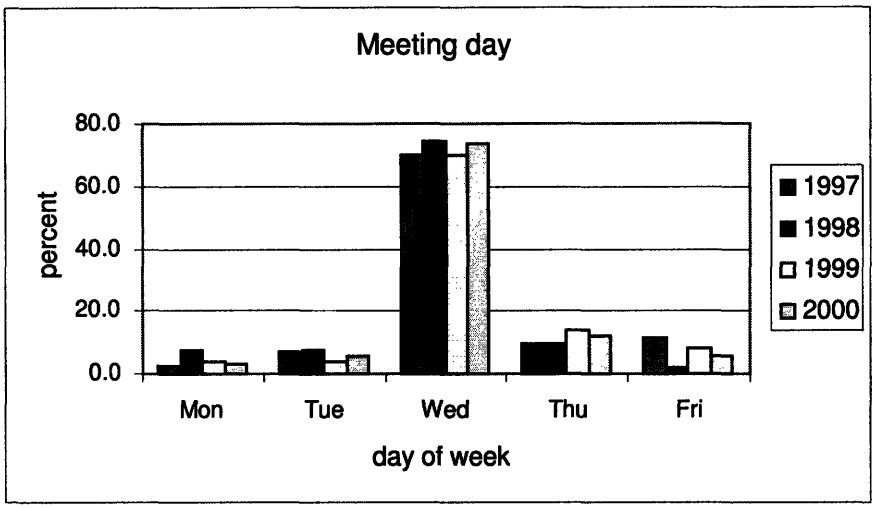
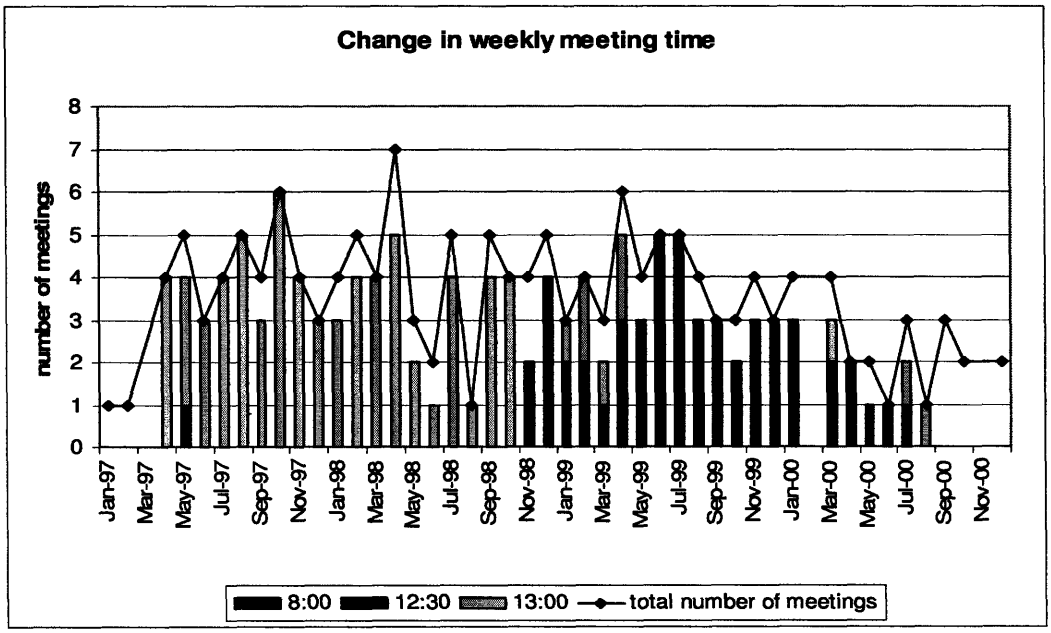


Figure 3.5 shows the proportion of the three meeting times over the four years. As seen in the graph, the four years may be broken into three phases where each of the three meeting times is dominant, serving as what I call "standard meeting time." A shift in temporal structure in organizations usually indicates other changes in organizations, and such was the case in the change of standard meeting time in LC. In the first phase (Jan 97~Oct. 98), the standard meeting time was 3 pm EST/ 1pm MST/ 12pm PST. 86% of

[19] In most cases, time/date of a meeting was available in the meeting minutes. For the exceptions that lacked proper time information, I identified them by matching members' phone records with meeting minutes. The LC meeting minutes recorded the meeting time very precisely, but I rounded it down to the nearest half hour in this analysis (for example, 1:08 pm was converted to 1:00 pm).

phone meetings during phase I happened at this time. This initial standard meeting time was established soon after members had started the weekly phone meeting and called "regular (meeting) time" by LC members. This meeting time persisted for the longest period of time, and was still referred as "regular time" even after the standard meeting time had changed. This time frame covers the majority of LC's product developmental period, from the start of the project to the release of the beta product.

Figure 3.5 Change in weekly meeting time



The brief second phase (Nov. 1998 ~ Feb. 1999) seems to be transitional. 63 % of meetings in phase II happened at this time. It was not accidental that they moved the meeting from the "regular time" to early morning. As noted before, Dan's new-born baby affected his daily temporal structure, which already had been complicated by various child care duties. Now he had a baby at home who constantly needed someone's attention

and care. Although Dan hired baby sitters on a regular basis, they were not available at all times and extending their service at the last minute was not always easy. After repeated delays and rescheduling of meetings caused by Dan, members finally decided to move the meeting to the early morning. At the beginning, however, this transition was not deliberate. Despite the repeated difficulties, neither Dan nor the rest of members actively sought to change the well-established, more than a year old routine of "regular time." The change was triggered by Fred's one-time schedule conflict. In attempt to reschedule the meeting, the early morning time was suggested as a possibility. Dan took this opportunity to inform others that early morning was better for him than the regular time. "Anytime this morning 10 or later is fine with me," wrote Dan in the email and added in parenthesis, "(In fact this is generally a better time than the 3:00 pm call, which can run on into the witching hour)." The other members acted upon Dan's nuanced request right away and added "find a new meeting time" to the next meeting's agenda list. As a result, they changed the standard meeting time to 10 am EST/ 8:00 am MST / 7:00 am PST. It was quite early for members in later time zones, but it was acceptable for Robert in MST who usually started his day very early and the two other part-time members in PST did not mind it a few times a month.

Nevertheless, the two part-time members missed a few meetings during that period, so LC members sought another meeting time that was not too inconvenient for those on the West Coast. Gradually, a new standard time emerged. Note that, in the later two months of phase II and the earlier two months of phase III, the traditional "regular time" of 1 pm in MST coexisted with the other two standard times. After a few experiments, members finally reached a consensus on the new meeting time, which was

only 30 minutes earlier than the traditional "regular time," but much safer for Dan in avoiding the "witching hour" and much more convenient for members at the other sites. In contrast to the previous one that only lasted for a couple of months, this new standard meeting time lasted almost to the end of the company's active history, comprising 73% of meetings in phase III.

The standard meeting time seems to have lost importance after mid-2000 when part time members left the company. The company-wide phone meeting became less important by then, since it was just a matter of coordination between the three full time members, who constantly communicated on a daily basis anyway.

The analysis so far shows that LC members enacted a stable, recurrent temporal structure around their weekly phone meetings. Maznevski and Chudoba (2000) found that successful virtual teams had strong, repeating temporal patterns. The weekly LC phone meeting is an example of such repeating temporal patterns which members introduced early and stabilized over time. In particular, the "regular time" of 1 PM in MST was so well established that members were reluctant to change the meeting time despite repeated failures to start meetings on time; even then, a few months after the change, it was changed back to one closer to the "regular time." This example suggests that a temporal pattern that has been established early and lasted a long time tends to be sustained even after it has ceased to be as functional as before, because changing it might well disturb individual temporal structures that used to have been coordinated around the long standing pattern. Anchoring an important organizational event with a specific clock time and a regular weekly cycle created a shared temporal map in LC. It provided explicit and regular opportunities for checking and realigning individual efforts. Moreover, LC's

"regular time" or "standard meeting time" represented the "appropriate time" for activities that required the temporal co-presence of all LC members, which guided the scheduling of any ad hoc meetings or events.

Conclusion

LC members' actual temporal practice does not seem to contradict much their perception that they had high level of flexibility in structuring their day. They had freedom to choose when to work or how many hours they work on a particular day. The amount of their work load was probably no less than that of average software engineers, but the flexibility they enjoyed in allocating their time according to their needs compensated for a long day. The LC members' estimate of a 50-hour week represents the temporal norms and standards in LC, the level of productivity and commitment that members deemed appropriate for highly skilled technical professionals.

For individual members who worked from home, flexible work first required them to find their own temporal rhythms and structures both to refrain from incessant work as well as to enable consistent work. Keith's daily walk, Robert's weekend garage sale visits, and Dan's habit of working late hours to avoid distractions are a few such routines that members created to structure, order, and regulate their temporal experience. It involved blurring boundaries between various activities that used to be separated into different realms of life, as well as establishing new ones. However, each member's sense of boundary was different, reflecting their family situations, local contexts, personalities, etc. In the absence of formalized temporal structures and hierarchical boundary control,

LC members had to achieve boundary management as group, which involved negotiating individual flexibilities through which members' different temporal expectations, interests, and patterns were made visible. Once members learned others' temporal patterns over time, much of boundary management was done through mutual adjustment without explicit coordination. Creating and establishing an organizational routine, such as LC's weekly phone meetings, facilitated coordinating flexible schedules of individual members by providing a temporal anchor in LC. Flexibility in LC was socially accomplished through mutual adjustments, negotiations, or sometimes conflicts among members.

Flexibility is generally believed to be a good thing, but it is never a free good (Golden and Powell, 2000). It took time and effort for LC members to learn how to put the idea of flexibility into useful practices. Because flexibility for one person can pose constraints to others, individual flexibility does not always benefit collaboration. In fact, this is the main reason why many software teams maintain long and rigid schedules, assuming that software development work itself makes them necessary. The dependencies between modules assigned to individual programmers are often pointed out as ground for the claim that they have to be present at the same time to be productive. However, the very fact that coding tasks can be modularized frees up programmers to work independently, and there is even counterevidence that allowing programmers to set aside a certain block of time to work flexibly actually improved productivity (Perlow, 1999). For a similar reason, software developers were found to work most flexibly among various technical contractors (Evans and Barley, 2004). What these seemingly contradictory findings suggest is that flexibility (or the lack thereof) does not reside in the

work itself. Rather, it derives from the innovation in social relations of the people who actually do the work.

Evidence from LC strongly supports that it is not the nature of the work but the *perception* of what it requires and the *practice* of how it is coordinated that need to be modified to facilitate flexibility (Perlow, 1999; 2001). It has been recognized that organizational members' sense of time is influenced by their work (Zerubavel, 1981) and that occupational differences in time orientation exist (Barley, 1988; Dubinskas, 1988b). However, as I pointed out above, different temporal norms and structures exist across groups and organizations that engage in the same type of work (e.g., software development). That suggests that research should extend its analysis to how work actually gets done and coordinated to properly understand the connections between work and temporal structures in groups and organizations.

In a similar vein, the influence of technological mediation may vary across groups and organizations. Recently, researchers have become increasingly aware of the complexity of the relationship between technological mediation and temporal practices in organizations (Steward, 2000; Tietze and Mussion, 2002). Moreover, individuals in the same group can experience time differently depending on their local context. Even for a small team like LC where members were very homogeneous in terms of task, expertise, professional position, age, and gender, individual members' temporal patterns varied by their different marital and family situations. In teams and organizations that have more variations in members' local contexts, coordinating these heterogeneous temporalities and achieving flexibility may become a more complex and difficult task.

CHAPTER 4

Coordinating through communication over distance

There was one thing LC members unequivocally stressed when they described their experience of working in a distributed environment: they were able to do it, quite successfully, because they had the capability of working independently on an interdependent project. And being a Ph.D. was an emblem that signified this capability, as revealed by Keith's comment below:

> I think that we were able to do it because we were Ph.D.s. I mean, and it's not so much the arrogance of being a Ph.D. or that we're better or smarter or anything like that, but you really are, we're Ph.D.s in programs where you really worked, where you really did your research independently. In other words, this wasn't, we weren't in a physics environment, we weren't, none of us has ever been on a paper with 35 authors, you know, or anything like this. We did very independent projects, when we were Ph.D.s. *We knew how to do that, we knew how to work alone, we knew how to collaborate with small groups of, very small groups of people.* That is what one learns when one's a Ph.D. that one really doesn't learn any other place (italics added).

LC members believed that they possessed not only the required technical expertise but also the necessary organizational and interpersonal skills to embark on a distributed software development project. Each had specialized skill sets to work independently on a part of the technology and all had enough experience in collaborating with a small group of people. A distributed project seemed to simply stretch what they already had been doing in their professional careers, using the skills they already had learned from elsewhere – coordinating individual tasks among a small group of collaborators and integrating them into a working whole, but this time, just from a distance. At least at the beginning, LC members did not see the distributed coordination

as much of a challenge to the project: "we weren't going to get them in one location, period!!" LC members believed they were competent enough to do a distributed coordination, at the same time taking full advantage of the flexibility it allows. Robert expressed a similar view:

> We had made our careers as doing our own thing, and the fact that we weren't living together was not really an overwhelming have-to-solve-this-problem thing.

In this sense, LC members' abomination of rigid work schedules I described in the pervious chapter was based on their self-image as experienced professionals competent enough to do otherwise—that is, to achieve flexibility in when and where to work by structuring and coordinating work activities more efficiently. If so, did it happen as they anticipated? Did LC members find the appropriate coordination structure to support the temporal flexibility highly valued in LC?

Hidden challenges of distributed coordination

Despite members' confidence, LC faced various issues of coordination from the onset of their collaboration. As I described in chapter two, LC members were not exempt from failures of dependency coordination. However, at least LC members had much experience in dealing with the dependency problems. Detecting and correcting those errors was more or less straightforward. The code would either work or "blow up," and once it blew up, members could locate the source of the problems with relative ease. Consider the following example. One day, Dan sent an email to the rest of LC members because he could not find certain files that his code was dependent on:

Date: Fri, 21 Mar 1997 00:23:00 -05:00
To: all
Subject: Missing files?

Is "EntityTree.proto" missing from ProgEntity? I asked because my make fails for lack of "EntityTree.scm," which is listed in INSTANCES in the ProgEntity Makefile.

dan

Robert replied very promptly, saying "My fault. It's there now."

However, the challenges of technologically mediated, geographically dispersed work went beyond detecting and correcting dependency problems. In fact, the failure of dependency checking originates from the failure of maintaining "mutual knowledge" (Cramton, 2001), since successful dependency coordination presupposes members' "awareness of tasks and the team" (Kraut et al., 2002), which is harder to achieve in a distributed situation. Moreover, misunderstandings caused by failures to maintain mutual knowledge are not always detected or corrected easily, often obscured by more apparent technical problems at hand. Still, they can result in serious consequences to the performance and cohesion of the team.

The conflict that broke out in the early phase of the LC project around the Linker module was one such case. Apparently, Keith and Robert had a disagreement about how to design the Linker module, which became particularly salient because this module significantly affected Keith's work at that time. The conflict escalated in just a few days to the point that it almost destroyed the collaboration. Robert wrote in an email to Keith, "I will not speak with you on the phone without another partner present. I am planning to take next week off to get away from things and evaluate my future involvement with LC." In response, Keith wrote:

I am truly sorry that you feel that way.

> I think that it is clear that we have different views of what is necessary to produce a product that meets a rapidly closing market opportunity.
>
> I know that you are strongly motivated by the personal satisfaction of "doing things right." You are in every sense a craftsman and I try to respect that. I get very frustrated because I cannot seem to ever get you to back away from those high standards. Sometimes, in the real world, you have to be able to back away.
>
> If you decide to leave, and I hope that you do not, your craftsmanship and technical prowess will be sorely missed; not only by me, but by the rest of us.

Robert came back a week later, and the entire team had a conference call. Although the conference call did not solve all the problems, it brought the team back together. Robert resumed working on the Linker, but tension soon developed again because there remained fundamental dissents regarding the module. Fred, who mediated between the two, talking back and forth, explained the difference:

> Robert and Keith, it turned out, had different ideas about...how robust the linker had to be. ...Robert wanted to do a technically and intellectually whole piece of software...whose design would survive in the face of what we all knew would be myriad challenges put on it, myriad tasks it would have to perform in the future. And Keith was just interested in getting something done that would enable him to continue with the development of the [LC system].

At issue were not only the technical details of the linker module itself but also how the entire development project should be approached and prioritized: "this was [the] issue of sorting out how we were going to work together," commented Fred. Once signing up for the task, Robert approached it very thoroughly which had been known as his style, while Keith expected a less sophisticated module that would be just enough to eliminate the bottleneck and allow them to proceed to do more critical tasks.

For a time, however, LC members did not seem to realize the significance of this difference nor the magnitude of its effects. In short, Keith and Robert failed to communicate the most important contextual information, the goals and assumptions each

brought to the particular module they were going to work on together. Without this common ground, continued discussion to resolve the difference only aggravated the situation, revealing more mutual knowledge problems that are typically found in other distributed teams (Cramton, 2001).

Uneven distribution of information: While the tension between Robert and Keith was growing, Dan also found serious problems with Robert's linker module that he had seen in his past jobs and that would affect his module severely. As Dan and Keith were working closely, talking on the phone on daily basis, Keith was also familiar with the problems, whereas Robert was left out of those conversations most times. Fred explained the situation in retrospect:

> Dan had found something in the linker that he had been bitten on before....Dan knew that this was going to be a problem. And Robert didn't...And this particular feature made Robert's design a lot cleaner and a lot more comprehensible, at least to Robert, and perhaps objectively, but Dan had been bitten by this before and he knew that this was a decision that would come around and haunt us if we just let it pass. And of course he had been talking to Keith who was more frustrated even than Dan over the whole issue, and they'd sort of been feeding off of one another.

Due to this uneven distribution of information, Robert could not properly understand Dan's problems, and asked in an email, "Why do I feel like I'm living on mars?"

Failure to communicate contextual information: Robert was busy improving the linker, but Keith and Dan were continuously experiencing problems with it. However, these problems did not hit Robert until he received a phone call from Fred, as Robert's email below reveals:

> I just got a call from Fred who had spoken to Keith
> and Dan and got the impression that they were spinning

their wheels because of the Link stuff that I wrote.

We talked several things to do about it:
1) I could spend some time writing a Link reference manual.
2) We could have a conference call to discuss this.
3) ???

what is needed?

Learning from the previous experience, Keith and Dan had Fred mediate between them, instead of directly expressing their frustrations to Robert. LC members agreed to have Robert write a reference manual, hoping it to provide a medium through which the points of disagreement could be laid out open and possibly resolved.

Difference in what information is salient: Different standpoints often hindered members' ability to understand the requests from other members and respond properly. In an email below, Keith is already expressing his exasperation about the difficulty of communicating what he really needed Robert to do:

I really do not want you to check this. I NEED to be able to loose bits. How many times do I have to say this!!!!

Unfortunately, Robert's continued attempts to satisfy Keith's requests only got him more agitated, and in the process, Robert's frustration also went as deep as to declare:

Keith, I refuse to participate in this stupidity..... First you wanted some way to distinguish names from names that a programmer could write. I gave you <name>. Then, you wanted to put function descriptors in symbol names. I gave you anything but quotations. Now you want quotations. Okay. Have quotations. I'm reserving the & as a delimiter. You can use anything else you want to for symbol names but it. Okay?

Now, all you have to do is come back and say "Oh, but I need to have ampersands in my symbol names, I really do."

Robert and Dan were facing a similar situation. Robert asked Dan to provide him a specific piece of information he needed, but Dan's answer was not satisfactory to him:

> Sorry to be so pedantic, but I thought my example was
> entirely clear enough, and I need to know what you thought
> was unclear.

I am now too pissed off to even try to look at this example.
I don't want to see this thing any more. All I want you to do is to tell me
which end of the field contains bit 2^0.

> I may see if I can scratch out the source code implementing this while
> the kids get their respective baths.

The last thing in the world that I want from you at this point is code.

Dan, still trying to understand Robert's request, sent another response, but it only fanned the flames further. Robert responded:

I will try this thing one more time. If I don't get an answer,
I'm going to be out of the linker business.
A valid answer is "I can't give you a valid specification."

Any answer which ambiguously renumbers the bits or includes
the letters PQRST will be translated as "I can't give you
a valid specification"

Although each member did his best to provide what the other wanted, without knowing why and how important it was to the person, it more often than not missed the mark and the repeated failures raised stress levels and even anger in both parties.

Difference in speed and timing / Uncertainty about the meaning of silence: Members also struggled with difference in speed and timing of feedback. Members took more time in responding to each other than normal, instead of promptly addressing sensitive issues. The difficulty of communicating nuances in emails also promoted silence to avoid more flare-ups. Nevertheless, when the tension was high, members more easily lost patience with each other and silence was often misinterpreted negatively as,

for example, personal neglect. For example, Robert complained in an email to Dan (copied to the rest of members), "I've asked Keith twice now for his algorithm [...], and he refuses to give me an answer. Do you know how he does this?" Keith saw this email and explained the situation:

> I did not refuse to answer your question.
> I did not understand the question the first time and by the time I saw the second notice, dan said that he was responding.

In the absence of mutual knowledge, ongoing discussion through emails only resulted in repeated failures to meet the others' expectations, which in turn seriously damaged interpersonal trust. At the culmination of the conflict, Robert cast serious doubts about the project and team, writing "my estimate of the probability of LC ever making any money is falling fast. It is far far below the point where I should really be out there looking for a job." At the same time, Keith was losing his faith too. As the direct consumer of Robert's linker module, its delay affected Keith the most. In this situation, Keith viewed Robert's meticulous approach as putting personal principles above the company's goal. Fred explained:

> Keith, sort of ran out of patience...he called me and....he [said that] Robert wasn't being very sensitive to the needs of the company in terms of moving forward in the development process...he was really frustrated...that this was an unnecessary diversion from what was going on, and he didn't understand why it was happening.

The reactions of Keith and Robert toward the behaviors of the other are consistent with Cramton's observation that failures of mutual understanding lead to dispositional rather than situational attributions concerning the cause of remote partners' behaviors or outcomes. This effect was inflated between Keith and Robert, who thought they knew

each other only too well. In fact, Keith and Robert used to work in the same company and lived almost next door to each other for years. The high expectation formed from the past combined with the relative lack of information on the other's current remote situation made these two members prone to dispositional attribution when their behavior suddenly diverged from their expectations. Fred also pointed out:

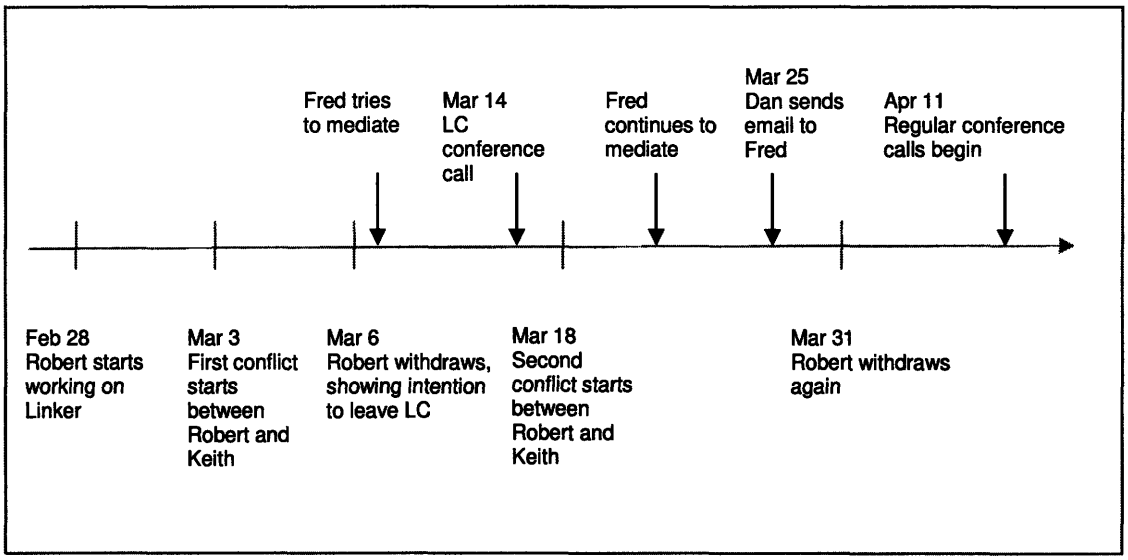
> This was the first time that there'd been any real contention in the company, and even just among these [three] people. And everybody was seeing a side of people that they'd never seen in the professional setting.

Laying blame on individuals, however, further directs attention away from analyzing and modifying the situation, which is almost always more complex than it appears to the remote partner. The discord and the discomfort accompanying it discouraged members from talking to each other directly on the issue, and it brought in more uncertainty in "reading others' minds." The email below where Dan is asking Fred's advice on a draft email illustrates this vividly:

> Fred, I was thinking of sending this to LC [group mailing list], but then decided maybe I should not. I am stumped as to what Robert is doing, or what he is up to, or if he is helping us get anything done at all, or if he intends to continue working for the company at all. I also no longer know how he is likely to react to anything, which is why I reconsidered sending this letter.
>
> And, I am really sorry to be asking your advice on this, this is certainly not what either of imagined we'd be doing in March.

Inevitably, performance of the team suffered during that time. The bottleneck that the linker module was supposed to solve was still there, while communication among the three full time members underwent rapid deterioration to the extent that it almost shattered the interpersonal trust, the very base of LC's dispersed collaboration.

Figure 4.1 Conflict around the Linker module



Ultimately, two noticeable changes came out of this contentious experience. First, members reflected on their email communications during the conflict and realized that they should be more careful in using emails. It was Dan who brought up the issue. With some humor, he wrote:

> From: Dan
> Subject: Analysis of recent e-mail exchange
>
> Our resident management consultant, who does not want to get in trouble, has been examining our e-mail. In this recent exchange, our consultant thinks it would have been better if you had taken a walk before replying to Robert's mail asking for a section reference. You were right, but you were insufficiently diplomatic about being right.
>
> Robert's reply, of course, also fails the diplomacy test, and also contains an embarrassing contradiction that our (otherwise non-technical) management consultant was able to detect with no assistance from our technical staff.
>
> Really, Martha thinks that we've got to do a better job with our e-mail exchanges, because we're having too many of these blow-ups.

At the same time, the difficulty members experienced in resolving disputes through email led them to seek for a better means to check possible misalignments of

expectations and goals among members. Suggested first by Fred and Robert, the weekly phone conference emerged through the conflict as a process intervention:

> Date: Tue, 01 Apr 1997 15:08:33 –0700
> From: Robert
> To: LC <all@LC.com>
> Subject: Technical discussions
>
> Fred and I would like to start having regular technical discussions.
> Fred proposed having the first one on thursday morning.
> Please send me a list of times that fit your schedule and any items for the agenda.

Eventually, learning from the early contention and conflict, LC became a more robust team. Above all, this early experience reaffirmed that they could not do without one another. They came to terms with their limits as well as strengths and learned to stay a team in the face of conflicts. As a result, the awareness of team and tasks in LC was much enhanced. Fred recollected:

> I was successful at telling...that we needed to solve this problem, but let's focus on solving this problem and not proving how correct you are...it was just everybody figuring out what the limits were [and] that there wasn't anybody but us five to mediate this. You couldn't storm into your manager's office and say, you know, my partner's being a real shithead about this, why don't you fix it. And the realization that you have to clean up your messes and that consequently you shouldn't litter the whole office all at the same time. And, there was never anything remotely as contentious as this again.

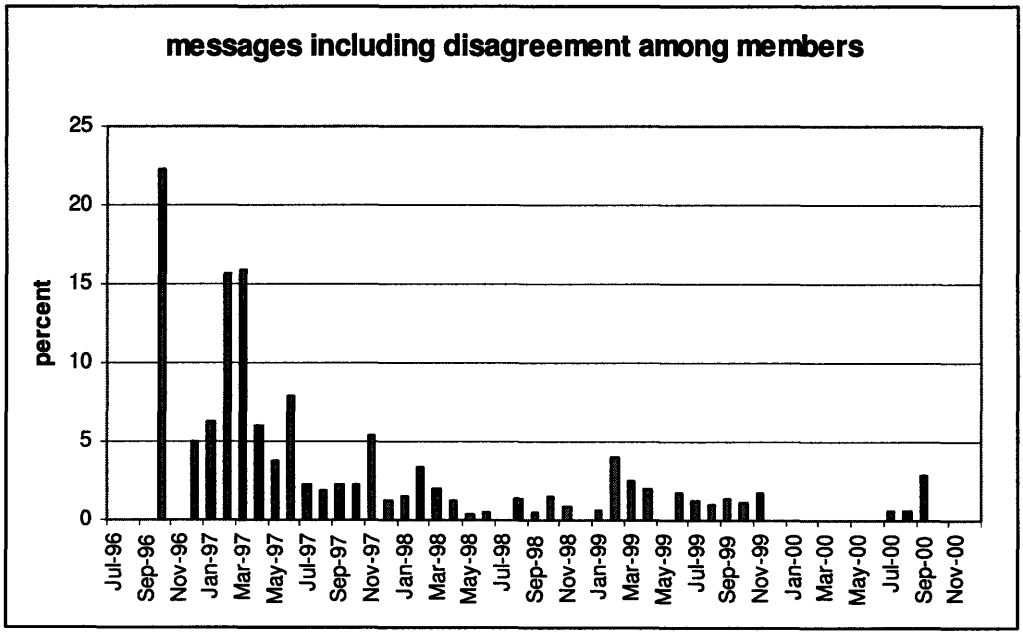
Fred was right in that LC did not experience conflicts similar to this again. Figure 4.2 shows the percentage of emails that voiced any disagreement between members. Messages including disagreements were concentrated in the first few months of the project and decreased thereafter. That does not mean, however, that the occasions on which members disagreed with each other suddenly became rare. Rather, it indicates that members indeed learned to use email more carefully and effectively. In emails, members

focused on presenting detailed information to support their arguments, avoiding overly contentious tone and language. Robert, who was in the center of the conflict, noted in his interview:

> We've had our stats. I mean, I fight with Keith quite a bit. Keith will often say we need to do this because of this and we need to do it this way. And initially, I would fight that if I didn't think that this was the right thing to do. What I've learned to do through the course of this company is to do a little bit of ground work to disprove, to show the fallacies in what I think he's saying. And sometimes, I'm successful when doing that. Sometimes he's right, in which case we'd go on and do something.

Also, instead of just arguing against each other through emails, members actively used phone meetings to sort out the differences together and find a compromise that would allow them to continue moving forward.

Figure 4.2 Percentage of email messages including disagreement among members



So far, I have shown that coordinating software development in distributed situation was arduous even for a small team composed of members who possessed a high level of knowledge about each other's skills, expertise, and personality, as well as strong trust formed from past collaboration. It was particularly so in the early phase of dispersed collaboration. There was still more to learn for LC members to coordinate effectively across distance, despite their rich experience of collaborating in small groups of professionals. Fortunately, through the conflict, members recognized the pitfalls early and took this opportunity to analyze, modify, and improve their communication and coordination practices.

Coupling coordination and communication

Coordinating distributed work required more than a spread-out version of good collocated work. LC members had to learn, through trial and error over time, how to maintain awareness of tasks and team. In distributed teams, communication is the key in creating, sustaining, and increasing such awareness. Successful distributed work seems to require a more innovative approach to communication and coordination. In what follows, I will analyze the ways LC members structured their communication to align and coordinate their distributed efforts from distance.

According to Perlow (2001), flexibility is closely related to the type of coordination used in organizations. In her study of software engineers in three different countries, Perlow found three different types of coordination: Managerial-centered coordination depends heavily on the relationship between the project leader and a group

of engineers, each of whom is a specialist; expertise-centered coordination depends on engineers supporting each other based on their areas of specialization; team-centered coordination depends on a group of highly interchangeable members with overlapping skill sets. Both managerial- and expertise-centered coordination depend on a personal mode of coordination whereas team-centered coordination uses a group mode of coordination.

The coordination in LC was more evenly distributed among the members because there was little hierarchical differentiation between members. Although officially Keith was the president of the company until they hired a CEO much later (beyond the time period covered by this study), Keith was more or less equal to other members. At most, he was the first among equals. The same was true in terms of coordination. Instead of having a central coordinator, each member evenly took up the responsibility of monitoring and coordinating the flow of work. Viewed from Perlow's scheme, LC's coordination type can be positioned between expertise-centered and team-centered coordination, closer to the latter. I will refer LC's coordination type as "expertise-based group coordination," as LC members used the personal and group mode of coordination dynamically in accordance with the situations – for example, the kind of expertise needed or the magnitude of the problem. Each coordination type was closely coupled with distinctive communication patterns and expertise coordination.

Personal mode of coordination – dyadic communication

In LC, the relationships and interactions between the three full-time engineers were defined not by an organizational hierarchy but by their relationship with the

technological system they were building. In other words, their role in the company was understood in terms of the interrelationship between the parts each of them was responsible for. This isomorphism between the structure of the team and the structure of their technology was then used to organize and further guide the work flows and interactions. Robert also used the isomorphism in describing their work and communication structures:

> You could draw a little diagram with Keith in the middle and me on one side and Dan on the one side. So, Keith did have lots of interactions with Dan, and I had lots of interactions with Keith. And Dan and I haven't really, we've worked on some stuff but, in comparison, we haven't had as much stuff that we worked on together..... This would correspond a lot with the way the product is organized because I've worked mostly on front end issues, reading the Java code and building. And Keith has done more of the kind of what's called the middle end issues of doing the optimizations of the compiler. And Dan has done most of the back end issues, which, this middle end, this is actually a technical term ---- of taking the optimized code in an internal form and turning it into the Pentium code that gets run by the machine. *And so, this diagram doesn't really say anything about Dan's personality and my personality as much as it says about just the way our world was organized in terms of the product* (italics added).

The project was divided into three big pieces, corresponding to each member's technical expertise. As mentioned in chapter two, LC could only be launched because Keith could bring together a team with three full time engineers who had complementary expertise and skills indispensable for designing and implementing the technology. At least in the beginning, therefore, it was natural for them to dissect the complex technology based on their preexisting specializations. Dan recalled how they decided on the initial division of work based on what each did best:

> There was a round of who had to work on what. Because I had the most experience with Code Generation, I ended up working on it. *So I could have worked on some other things. But no one else had worked on Code Generation.*

> Keith had more experience with attribute grammar. I felt that that was the way to work the two-level language. So he ended up doing that. And I think Robert ended up writing the first natural language parser. That was a case of specifications which Robert was very good at (italics added).

The initial division of work based on each member's specialization further consolidated the isomorphism between the structure of technology and the structure of work. It allowed them to work on their tasks on an individual basis, lessening the burden of simultaneous interactions and creating room for temporal flexibility. However, this situation also created a high need for close collaboration for the two whose tasks were closely related. Robert explained in the interview:

> In terms of technical things where we really didn't do technical things as the whole company, for the most part, it was like we made decisions as to which technical things were important but actually hashing out how we were going to do something and how we were going to solve the problem, improve performance in a certain area. That was more done on an individual basis or a one-on-one basis with, I mean Keith.

On a daily basis, each LC member was responsible for his assigned tasks and coordinating with each other to ensure compatibility. As a result of the dependency structure, dyadic coordination was often needed between two members whose parts were joined up. Whenever a problem arose or they were not sure how to proceed, they turned to each other for help, using email and/or phone.

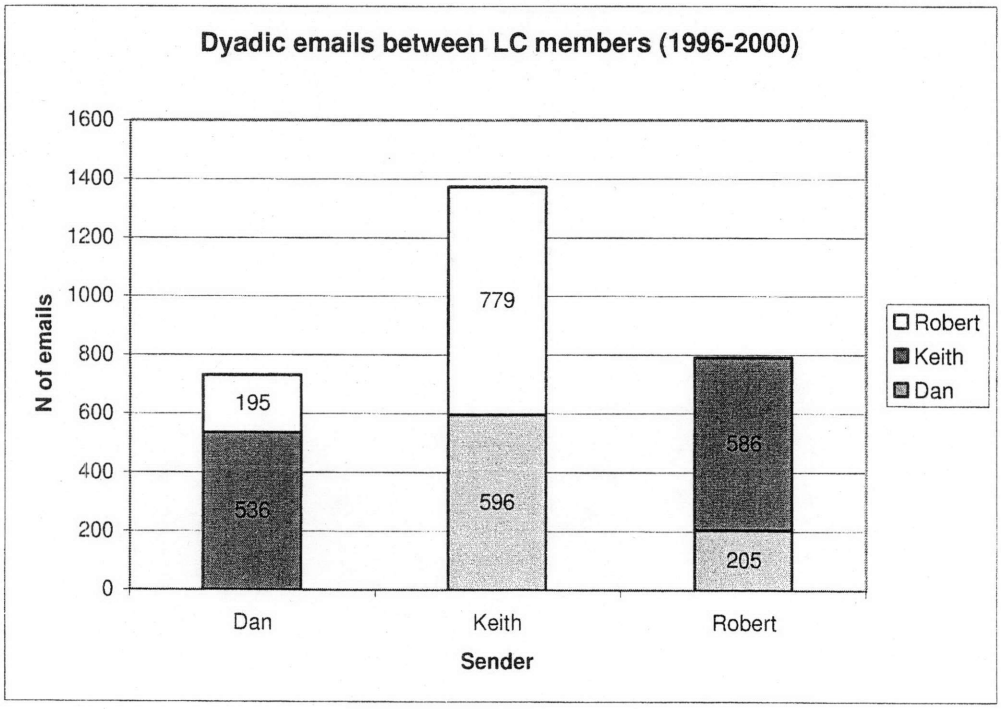
The actual communication pattern in LC supports this. LC members sent on average 24% of total emails to one of the LC members. Furthermore, the pattern of dyadic emails is consistent with Robert's description of isomorphism between the structure of the product and the structure of work interactions. Table 4.1 shows different patterns in dyadic emails among the three full time members. Dan had almost 3 times as

many dyadic emails to Keith as to Robert (13.4% vs. 4.9%) and Robert also had almost three times as many to Keith (17.3%) as to Dan (6.1%), whereas Keith had only slightly more dyadic emails to Robert (21.5%) than to Dan (16.5%).

Table 4.1 Email communication among LC members (1996-2000)

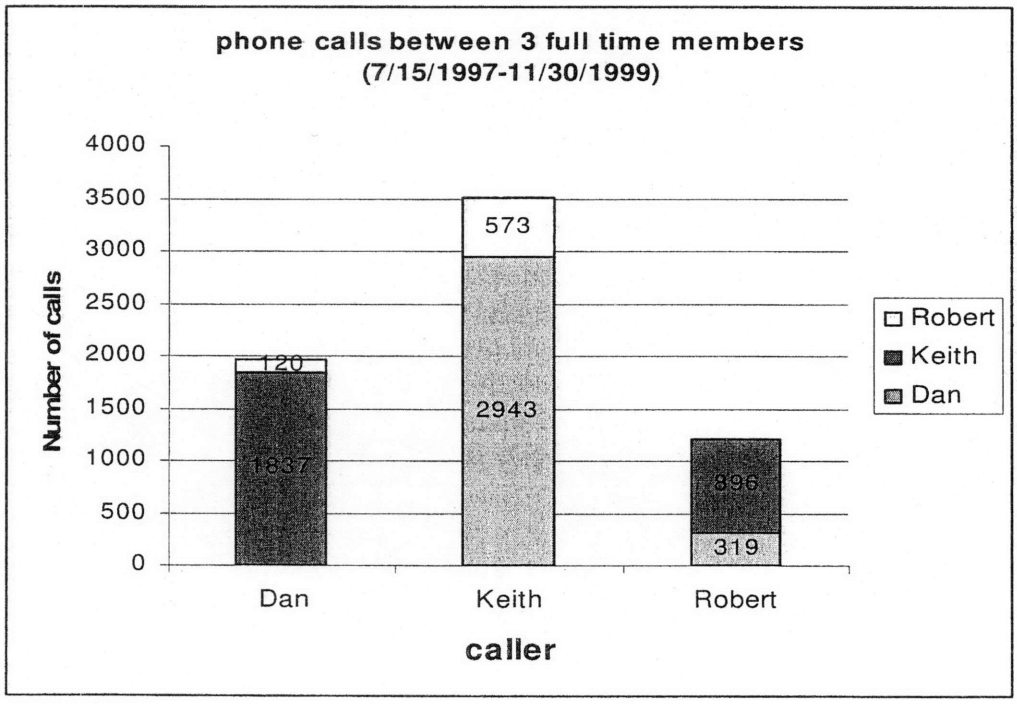
Receiver / Sender	Dan only N (%)	Keith only N (%)	Robert only N (%)	Fred only N (%)	Martin only N (%)
Dan	-	536 (13.4%)	195 (4.9%)	44 (1.1%)	23 (0.6%)
Keith	596 (16.5%)	-	779 (21.5%)	5 (0.1%)	2 (0.1%)
Robert	205 (6.1%)	586 (17.3%)	-	48 (1.4%)	39 (1.2%)
Fred	40 (11.5%)	6 (1.7%)	43 (12.3%)	-	1 (0.3%)
Martin	24 (4.1%)	1 (0.2%)	28 (4.7%)	0 (0.0 %)	-

Figure 4.3 Dyadic emails between three full time members (1996-2000)



The pattern in dyadic phone calls reveals a similar but slightly different pattern (Figure 4.4). Among the dyadic phone calls each full-time member made, 88% of Dan's and 60% of Robert's calls were made to Keith, whereas calls from Dan to Robert and the reverse were only 5.8% and 24.6% respectively. This is similar to their email pattern in terms of the centrality of Keith in their interactions. However, in contrast to Keith's email pattern where he sent slightly more emails to Dan than to Robert, Keith made more than 5 times as many phone calls to Dan as to Robert (59.8% vs. 11.6%).

Figure 4.4 Dyadic phone between three full time members (1997-1999)[20]



[20] As mentioned in the data section, the phone records provided by three full time members cover different time periods. To compare the dyadic communication patterns between sets of three full time members, I only analyzed the data for the period when the phone records of all three full time members are available.

This imbalance originates from individual differences in media preference and mutual accommodation of members to the differences, as the following statements from different members corroborate:

> [Keith] And so we either do email or we talk on the phone, and I will tend to talk on the telephone more and Dan will tend to talk on the telephone, or Robert really prefers email, and so this is little sort of twister, but we've grown comfortable with all of this. It's just a compromise. He [Robert] will just generally send a piece of email and I'll call him back on the phone [Laughter]. But this doesn't mean he never calls on the phone or anything like that. In other words, there are extreme places over here where you have to use the telephone and there are extreme places over there where email is clearly the right answer, and there's a big middle ground. And what I am essentially saying is that Robert is going to put it over on the email side, [he] is going to sit on the side of the email thing and I'm going to sit on the side of the telephone.
>
> [Robert] I fought Keith a lot because of just over this issue. I wanted to do more [emails] and I still want to do, although I've kind of gone to using the telephone because Keith works better with that, but I tried to get most of the stuff on email.
>
> [Dan] As a practical matter, I felt most comfortable on the phone talking to Keith.
>
> [Fred] Robert definitely preferred email. When I needed to call him and ask him about something, it didn't bother him to talk on the telephone or anything like that, but I think that Robert just thought that he organized things and didn't waste as much time if he used email. And Robert's not quite as chatty as Dan, or Keith, or I, for that matter. And Dan and Keith, I guess I wouldn't say that they had a preference either way.

Robert's strong preference of email over phone as main communication method is evident. As the Figure 4.4 shows, Robert used phone the least among the three. Knowing Robert's preference, the other two members used more emails than phone to communicate with him.

The personal mode of coordination in LC appears to be similar to what Perlow (2001) called expertise-centered coordination, "where members depend on each other to coordinate the work based on their areas of specialized knowledge (p. 104)." As Perlow

pointed out, expert-centered coordination can create more pressure for overlapping hours because members cannot fill in for each other when a specific individual's expertise is indispensable to complete the work. In LC the three full-time members initially divided the whole project among themselves based on their expertise, with virtually no back-up personnel to turn to. In fact, it is not difficult to find evidence that members sometimes needed the expertise of a specific individual and asked him to be present for interactions. For example, members often sent emails to request help from a specific member:

> Date: Wed, 17 Jun 1998 10:06:16 -0600
> From: Robert
> To: Keith, Dan
> Subject: help
>
> When you guys get off the phone, I need Dan's help.
>
> I'm crashing after I get a NativeAllocate of size
> -601917076 and need help finding out why.
>
> robert

Also, Keith often sent an email with just a subject line to request a phone call, as a few examples below illustrate:

> From: Keith
> To: Dan
> Subject: are you working tonight, if so please call me.
>
> From: Keith
> To: Dan
> Subject: if you are still awake, call me.
>
> From: Keith
> To: Robert
> Subject: Call me when you get a chance.

At least in the specific areas where one member needed the other's expertise to complete his own work at hand, a personal mode of coordination using dyadic emails and phone calls was useful. The kinds of problems that were specific enough not to affect the

third member directly were mostly handled in this way so that they would not interrupt others' work schedule more than necessary.

Group mode of coordination – group communication

The personal mode of coordination is effective in coordinating between a subset of members, minimizing interruptions and information burdens on the other members whose work is not (or is predicted not to be) directly affected by the coordination activities. However, when the personal mode of coordination is dominant, it can generate mutual knowledge problems, particularly uneven distribution of information among team members. I already described the episodes around LC's early conflict, in which Robert did not have equal knowledge about the problems that Keith and Dan had already discussed together. Given the basic division of work and communication patterns that promoted more communication between Keith and each of the other full time members, it might have created a situation which forced Keith to take up the role of central coordinator to fill the gap between the two other members.

Instead, much of the work was coordinated collectively as a group in LC. For example, not only was the decision to assign what to whom made as group, but also everyone was informed if one of them ran into problems in his task. Even when a problem had already been handled by any two members, the rest were acquainted with the process and result soon after. As a result of such frequent group interactions through emails or phone, LC members had a great sense of what each was doing and what problems each faced. Ray, the CEO who was hired by the team in 2001, pointed it out as the most distinctive team dynamic of LC:

> It was an eye-opening experience, to be honest with you. They don't do anything without the other person knowing it. Every email that was sent to the company goes to everyone in the company. If they have a problem with a piece of code that Robert wrote, everyone in the company sees it.

The central role of group coordination in LC is also corroborated by their communication pattern. ***Group email*** that was sent to every member was the most basic and daily means of group coordination. As seen in Table 4.2, LC members sent out over seventy percent of their emails to the whole team, except for Keith who had a high proportion of dyadic emails because his work was closely related to both Dan's and Roberts' and thus required more frequent dyadic interactions. Through group emails, members shared critical information about tasks and team as a group, without constantly interrupting others as members could read and respond to the emails at their convenience.

Table 4.2 Distribution of dyadic and group emails in LC

Receiver / Sender	Dyadic email		Group email	
	N	%	N	%
Dan	797	19.9	2963	73.9
Keith	1382	38.2	1992	55.0
Robert	878	25.9	2466	72.4
Fred	90	26.8	246	70.5
Martin	53	9.0	519	88.0

Group phone meetings are another important example of LC's group mode of coordination. As described previously, the decision to have a group phone meeting on a regular basis emerged in the course of trying to resolve a conflict. Weekly phone conferences became an organizational routine quite early and remained central throughout the collaboration. Typically running from forty five to ninety minutes, these phone meetings were almost the only means of synchronous group communication in LC.

During these meetings, members ensured that they had a symmetrical understanding of important issues, filling possible holes in dyadic communication or group emails. Fred's statement below demonstrates this point:

> Well, the weekly conference calls were sort of like staff meetings, I mean. Basically we were on the telephone all the time, but not all five of us together...It was just five guys, and we all knew each other pretty well and we had all been talking to one another during the week. So there wasn't normally a lot of "status" to review at one of these conference calls, *it was just our mechanism for making sure that everybody was kept in the loop, and that not too much business got done on these two-way or three-way phone calls that nobody else found out about until it was too late.* And so we, for a while we tried to set up specific agenda items...I tried to set up a template that would say what we had to discuss. It wasn't so much that we needed to spend so much time on any particular subject in the meeting, but just that, on these certain topics, anybody who had something new, that had come up during the week, had to make sure that everybody in the company understood what those were (italics added).

Given its purpose, assuring the full participation was of utmost importance. The attendance record in Table 4.3 shows that LC members were quite successful in having every member present in the meeting. Out of 181 meetings over the 4 year period, 72.4% achieved full participation.[21] In particular, the three full-time members boasted on almost perfect attendance rate across four years. Overall, the full-time members had a higher presence in the meetings than the two part-time members, reflecting their centrality to the team. Figure 4.5 shows the change of attendance rate by members over time more clearly. In 1999, the perfect attendance rate decreased due to Martin's frequent absence, but it

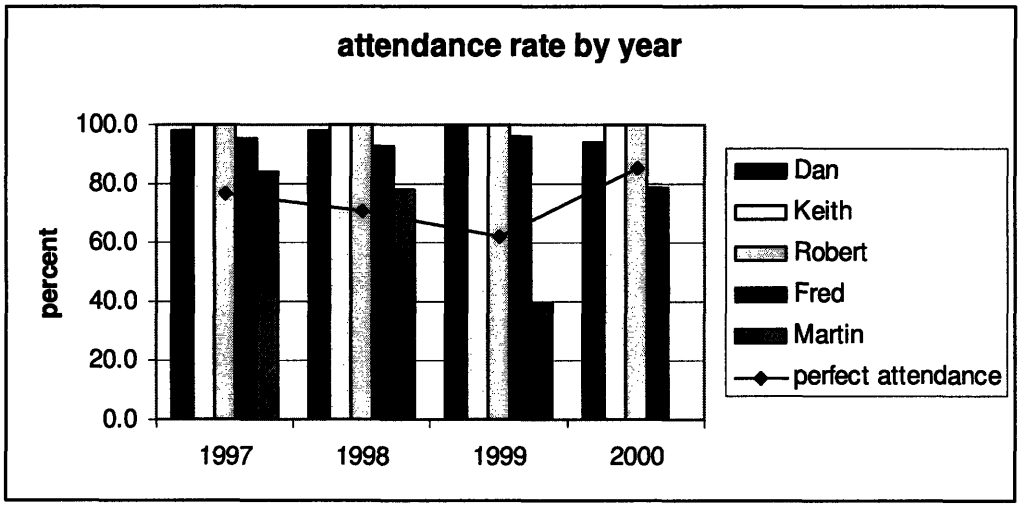
[21] The analysis is based on the total 181 meetings held between Jan 1997 to Dec 2000. The three full-time members participated for the entire period, whereas part-time members participated only a part of the period until they left the company (Fred until April 2000 that covers 171 meetings and Martin until July 1999 that covers 125 meetings, respectively). The attendance rate is the number of meetings where a member was actually present divided by the total number of meetings held during which he was still a member of the company.

went up in 2000, mainly because the participants decreased to the three full-time members.

Table 4.3 LC member's attendance in phone meetings

Member	1997	1998	1999	2000	Attendance percentage
All five	33	38	31	29	72.4%
Dan	42	53	50	32	97.8%
Keith	43	54	50	34	100.0%
Robert	43	54	50	34	100.0%
Fred	41	50	48	11	85.7%
Martin	36	42	11		71.2%
Total Number of meetings	43	54	50	34	-

Figure 4.5 Attendance rate across four years



As meeting time was limited, it needed to be spent efficiently. It did not take long before members recognized that it was necessary to structure their multi-way phone conversation for an efficient meeting. In his interview, Keith also pointed it out:

> If we don't structure it, we will get on the phone and just have an ad hoc unstructured meeting. But, for the Wednesday meetings that we've had, we learned very quickly that, if we didn't structure those, there was ramble and rant.

One of the structures members adopted was announcing the agenda and turn-taking upfront before the meeting. After discussing each agenda item, members would report briefly on their work status in turns. However, as Fred pointed out in his interview, members generally had fairly good knowledge of the others' status through frequent emails and phone calls already, so status reporting was usually brief, just enough to ensure everyone had parallel knowledge regarding the progress of the project. Members rather utilized the meeting to focus on the issues that involved key negotiations or urgent decisions, the kinds of interactions that, as one member told us, were "hard to do over email." Fred also voiced a similar opinion:

> Email was useful when you wanted to exchange a lot of technical information. It was really useful for things like position papers when you're trying to hash out strategy for building a product. But in terms of how we made decisions and stuff like that it was all conference calls.

The weekly phone meeting was an important means for coordination through which members aligned their individual work efforts on a regular basis and made important decisions. It helped members achieve an increased level of awareness and temporal symmetry.

In LC, the group mode of coordination could be dominant, because all three full-time members had not only complementary skills but also overlapping expertise enough to understand the others' work. As Keith put it, they all knew each other's own expertise, but it was not "territorial":

> From a technical point of view, each of us could really grasp. That doesn't mean we saw every little point exactly the same. But each of us really could grasp what the whole thing had to look like in order for this to work. And we were all experienced enough so that we knew what those pieces were and we knew how those pieces were going to have to interact.

The team's broad base of shared expertise was particularly important for making the group mode of coordination effective in LC. "Creating mutual understanding," Hightower and Sayeed (1995, p. 43) pointed out, "require[s] group members (using computer-mediated communication) to transmit much more information than those working face to face." When there is little shared expertise among members, it takes more time understanding the transmitted information. The broad base of shared expertise among members thus contributed to the group mode of coordination and communication, without limiting productivity by increased information burden.[22]

Interweaving multiple modes of coordination and communication

As seen so far, LC members used multiple modes of coordination, interweaving different media available to meet their everyday coordination needs. For technical tasks that needed the expertise of and close co-work with a particular member, they used the personal mode of coordination through dyadic communication. For tasks that did not need a particular person's expertise or that affected everyone, they used the group mode of coordination, through group emails and phone meetings.

---

[22] For the same reason, it played out differently with a member who did not possess adequate expertise to read and interpret information. For example, members also sent the group emails to Ray when he was hired for marketing, but stopped it soon upon Ray's request. For Ray, who did not have adequate technical background, the flurry of group emails on technical issues only clogged his email box, making difficult to sort out messages directly relevant to his work.

However, switches between the two types of coordination were also frequent as the situation changed. For example, the progress of tasks mostly handled by a subset of members was soon reported to the team through group emails and a phone call sometimes led to a three-way conversation when the third person's input was thought to be important. In reverse, a group email informing about a bug often triggered dyadic communication between those who were trying to fix it. Daily coordination in LC was characterized by the dynamic interweaving of different types of coordination using multiple media, as Fred's description below illustrates:

> So I would talk to Keith or I would like to talk to Dan, and if we didn't know something about the issue that was being discussed, then we'd get somebody else on the telephone. Or Martin and I would talk about how we were going to run the presentation to [company name], and then we'd get to the point where Oh God, we really need a couple points to, on this issue, and we'd call Dan, 'can you write a couple slides and email them to us?' and that sort of thing. And of course, after we made any sort of presentation, then we would, for the most part, have a conference call with everybody and sort of debrief them and tell them what happened, and if there were any action items from the meeting, like they wanted to see some particular figures or particular benchmarks or they wanted to be put on the list for the next Beta release or something like that, then we'd all sit around and talk about how that was supposed to happen.

A close look at how their actual work day progressed displays a more vivid picture. Table 4.4 summarizes the interactions between members on a single day, where all four combinations of coordination/communication mode can be found: personal coordination through dyadic emails or phone, group coordination through group emails, and group phone meeting. On that day, there were three main technical problems, two of which were handled through the group mode of coordination and the third through the personal mode of coordination. The two incidents of group mode of coordination were both triggered by a group email announcing a bug or error, for which other members

provided assistance, also through group emails. The third problem was in Dan's area and affected Keith's module, and frequent dyadic communication between the two continued throughout the day. In dyadic communication, members tended to use phone and email for different purposes. Emails were used to exchange stack traces or codes back and forth, while the phone was used to discuss the problems. Phone was more convenient for asking a simple inquiry, checking the status, and getting immediate feedback. As evident in Table 4.4, Dan and Keith exchanged short but frequent phone calls during the day, almost simulating interactions in a collocated office. There were also additional group emails notifying all of changes in tasks and local situations. LC had a weekly phone meeting on that afternoon too. This example shows that LC members used layers of coordination, coupling them with different media according to the task and situation.

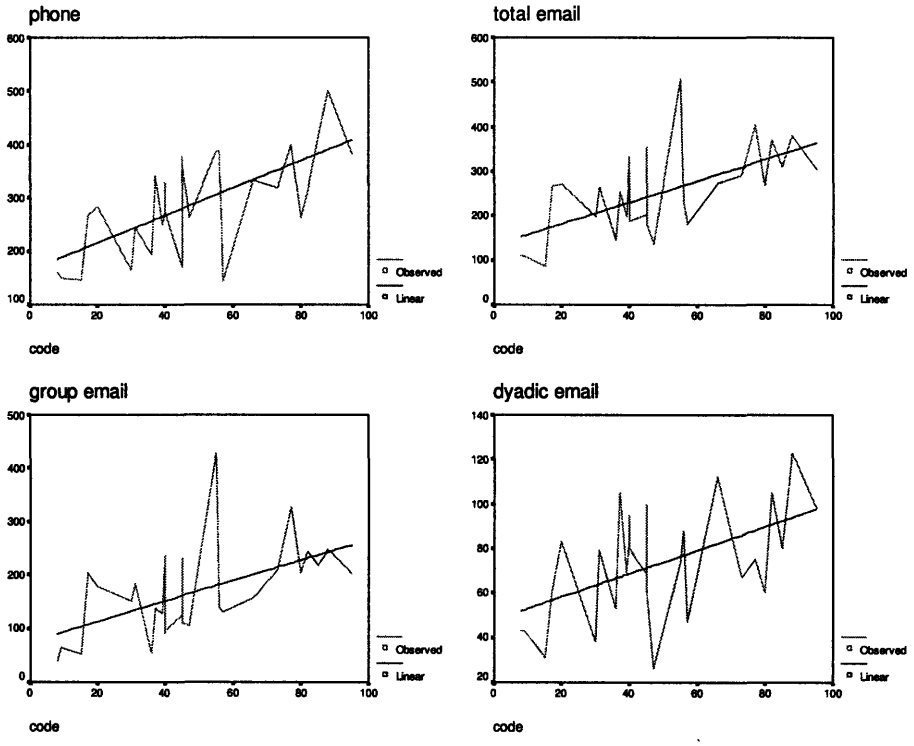
A further examination of LC's communication data reveals some additional patterns. First, LC members increased the frequency of communication of all types in periods of high activity. In Figure 4.6, I have shown the number of code commits on the x-axis and frequency of communication on the y-axis for four media types during 28 months from August 1997 to November 1999. All four types have a positive correlation with the number of code commits. Code commits, checking in a new working version of the system on the team server, had implications for the others due to the complex interdependencies among the modules, and thus needed to be carefully coordinated. "If you committed something, you gotta tell them that you did it. And you gotta ask them if it is OK," stressed Dan in an interview. Code commits therefore demanded a higher degree of personal and group communication for effective task coordination.

Table 4.4 A work day in LC

Time (in EST)	Media	From	To	Subject line of email or Conversation length (minutes)	Type of coordination*
9:08 AM	Email	Dan	Group	ooops	G
9:16 AM	Phone	Keith	Dan	16	P
9:32 AM	Phone	Dan	Keith	4	P
9:38 AM	Email	Keith	Dan	cg25	P
9:46 AM	phone	Keith	Dan	1	P
9:54 AM	phone	Robert	Keith	13	P
10:02 AM	email	Robert	Group	re: ooops	G
10:29 AM	email	Dan	Group	re: ooops	G
10:37 AM	phone	Keith	Robert	1	P
10:42 AM	phone	Keith	Dan	1	P
11:24 AM	phone	Dan	Keith	4	P
11:32 AM	phone	Keith	Dan	5	P
12:20 PM	phone	Dan	Keith	1	P
12:23 PM	email	Martin	Group	io problem	G
12:39 PM	email	Martin	Group	io problem -- may be my bug	G
12:43 PM	email	Martin	Group	io problem -- the original problem stands	G
12:43 PM	phone	Keith	Dan	6	P
12:46 PM	email	Keith	Dan	here it is	P
12:48 PM	email	Robert	Martin	re: io problem	P
1:10 PM	phone	Dan	Keith	3	P
1:18 PM	phone	Keith	Dan	1	P
1:19 PM	email	Keith	Dan	dead cg25	P
1:37 PM	phone	Keith	Dan	3	P
2:05 PM	phone	Dan	Keith	1	P
2:16 PM	email	Dan	Group	this afternoon's phone call	G
2:47 PM	phone	Keith	Dan	1	P
2:56 PM	email	Fred	Group	Alpha's system	G
3:07 PM	phone meeting	All present		82	G
3:26 PM	email	Fred	Group	exception in [...]	G
5:03 PM	email	Dan	Group	code committed.	G
5:08 PM	phone	Robert	Dan	1	P
6:02 PM	phone	Keith	Dan	1	P
6:33 PM	phone	Dan	Keith	10	P
7:30 PM	phone	Dan	Robert	10	P
8:57 PM	Phone	Keith	Fred	1	P
9:35 PM	Phone	Dan	Keith	1	P
9:45 PM	Email	Keith	Dan	i am back from the neighbor's house	P
10:48 PM	Phone	Dan	Keith	12.9	P

G: group mode of coordination, P: personal mode of coordination

Figure 4.6 Communication frequency and code commits



P < .01

Table 4.5 Correlations between combinations of communication and coordination types, 3 full time members

	Group email	Dyadic email	Phone	Phone meeting
Group email	1	.433 † (.021)	.621 ‡ (.000)	.506 ‡ (.006)
Dyadic email			.744 ‡ (.000)	.308 (.111)
Phone			1	.271 (.163)
Phone meeting				1

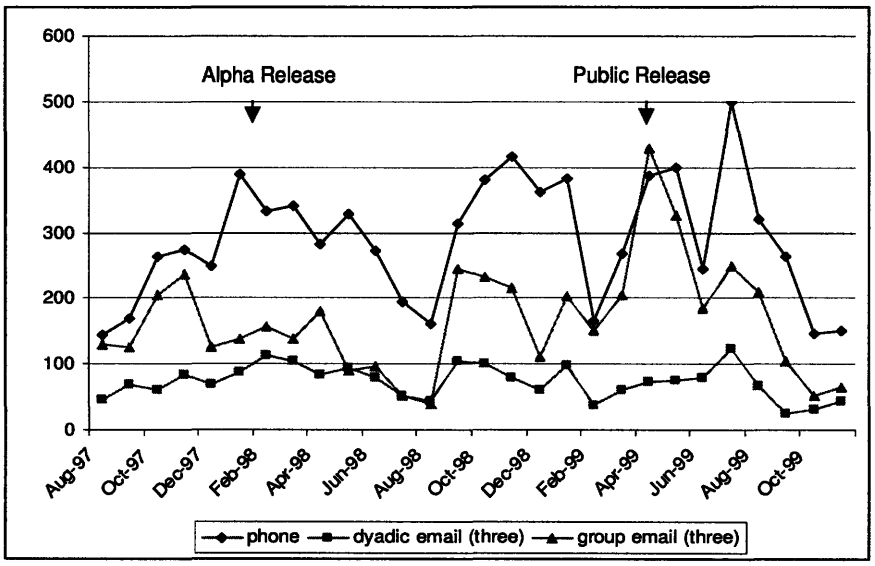
† p < .05 ‡ p < .01

In addition to adjusting communication in response to work loads, members seem to have learned to balance the two modes of coordination. I analyzed how each type of coordination co-varied over time during the same time period (Table 4.5). The result shows positive correlations not only in the same coordination type but also across different types. It is not surprising to see dyadic emails and phone calls increase or decrease together with the strongest positive correlation, as one tends to trigger the other when two people work on a task together simultaneously. But there are also positive correlations between group email and dyadic email or between group email and dyadic phone call. This reconfirms that members increased the frequency of communication of all types in periods of high activity. Moreover, LC members communicated more frequently as a group when they did so in dyads, which allowed members who were not directly involved in the dyadic coordination to maintain mutual knowledge about the task situation.

Members learned over time how to combine different modes of coordination and communication dynamically. Figure 4.7 depicts the patterns of communication in phone, dyadic email, and group email, and all three types of communication show similar trends. However, a close look also reveals differences in the relative use of the three types. For example, right before LC's alpha release in February 1998, the use of phone increased disproportionately, in contrast to the gradual increase in emails. Under time pressure, members used phone more to speed up the coordination process through immediate feedback. Then after the public release of the product in Apr. 1999, use of group emails shows a big increase, to a level almost as frequent as the use of phone. The technical tasks were reduced to bug fixing and tests, which by then required less coordination as

members had developed a fairly good sense of "to whom the problems belong." More importantly, the main activities of the team at that time related to marketing of the product, the area in which none of the LC members had much expertise; thus, it required more group communication and coordination.

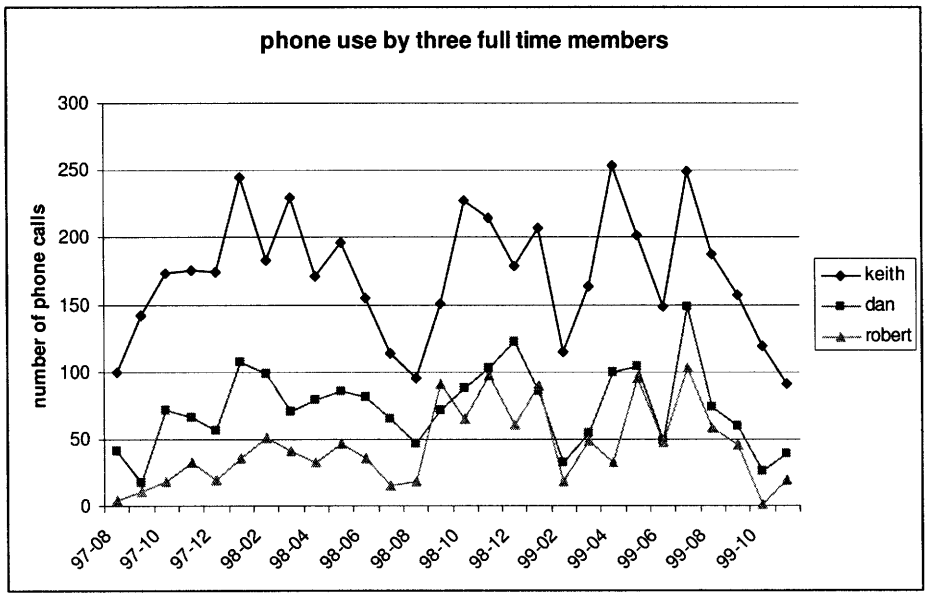
Figure 4.7 Communication pattern over time (three full time members)



This practice of increasing communication of all types and dynamically coupling communication with the kind of coordination needed appears to be the outcome of gradual learning. A good example is the use of phone in LC over time. The increased use of phone that emerged during periods of high technical activities in mid 1998 stayed on once members leaned its effectiveness for coordination. However, individual members adopted this practice at different paces. Figure 4.8 clearly shows that Keith always used the phone most among the three, Robert the least, and Dan in the middle. Whereas Keith and Dan increased their use of phone relatively early and stayed with that pattern in

general, Robert's pattern shows a slower adaptation over a longer period of time. Robert's use of phone was very marginal at first, but it gradually increased over the year, and finally became almost equal to Dan's by the fall of 1998. Dan also hinted in the interview about Robert's adaptation, commenting "Robert got much better with phone." Robert's strong preference for email may have slowed down the change, but ultimately he was the one who adapted his communication pattern most for better coordination, particularly with Keith.

Figure 4.8 Phone use by individual member over time



Despite individual differences in communication styles and preferences, all LC members seem to have learned from experience that "heedful interactions" (Weick and Roberts, 1993) are the best solution for mutual knowledge problems and gradually developed their own way of interweaving various types of communication and coordination. It also resonates with the findings from Ghosh et al. (2004) that emergent

communication norms were particularly dominant in LC over those established formally and explicitly.

Coordination, versatility, and flexibility

What are the implications of LC's dynamic coordination for temporal norms and practices, in particular, that of flexibility? One insight from Perlow' study (2001) is that coordination type is related to how expertise of the team is constituted and it in turn affects the temporal organization of work. For example, Perlow suggested that the personal mode of coordination, based on engineers with distinctive areas of specialization, creates a need for overlapping hours because they cannot substitute for each other. As the result, a relatively inflexible schedule is required to guarantee overlapping working hours. In contrast, the group mode of coordination is found to allow much greater variability in work time, because engineers do not depend on specific individuals to complete the work.

In reality, it is more likely that the coordination type in organizations shows variations from Perlow's typology, with more complex consequences to temporal norms and practices. LC is an example. Personal and group modes of coordination coexisted in LC, but the latter prevailed, facilitated by the horizontal team structure and overlapping expertise among members. Indeed, the positive thread between group mode of coordination, versatility of members, and flexible structure is identified in LC. The shared expertise was the key to this flexibility, since it allowed the team to assign work more dynamically according to the work demands and situations. Robert's comment supports this observation:

> We didn't really fight all that much about who does what. It was pretty obvious. There were a couple things that were just kind of there, and somebody had to do them. And the person who had the least other stuff to do did them, and a number of things that I worked on were like that.

The story of how Robert came to work on the "Linker" module provides an example. Initially, the linker module was assigned to Fred, one of the part-time members of LC, because he was most familiar with it. However, illness and other responsibilities prevented him from starting the work on time and created a potential bottleneck in the workflow. All of the three full-time members were equally knowledgeable with the module, but none of them had superior expertise over the others. Robert volunteered for the task because he was the only one available at that time while the other two were occupied with their current tasks:

> Date: 1997-02-28T22:30:37 GMT
> From: Keith
> To: Robert
> CC: Dan
> Subject: Re: Testing a garbage collector
>
> I just talked to Fred. He still sounded pretty sick.
>
> He has not started on the linker. Furthermore, his pc also seems to be sick.
>
> I have been hacking on the spec as I have been working on the assembler and my guess is that if someone does not start on it, it will be the bottleneck. In some sense this might be good for you get your hands in because as time goes on and we get sharable images, most of this will have to be tightly specified between the linker and the loader.
>
> keith

"Learning by doing" gradually transformed Robert into the team's expert on this module. Naturally, all subsequent linker problems were passed to him, as Dan mentions in the interview:

> When we said it was a Linker problem, it ended up in Robert's lap because he worked on the Linker to begin with and that's what he knew the best.

The overlapping skills enabled LC members to assign work more flexibly, and this flexibility in turn increased the versatility of members, which further reinforced the foundation of group coordination. It changed the context of expertise-based coordination as well, but it is important to note that this newly defined expertise emerged in the process of group coordination. The story of how Robert ended up working on the Garbage Collection (GC) module provides even stronger evidence for this point. If LC had to depend sorely on expertise-based coordination, the GC work would have been assigned to Dan, who was the best expert on GC in the team. However, Robert took up the work because Dan was busy with another important module. Robert explained:

> And Dan, his specialty was, well, he wrote his thesis on garbage collection. It's kind of ironic that I did the garbage collection work in this, although Dan has done a lot of follow-on help...I did the garbage collection not because I had the expertise but because Dan had all of this other stuff to do.

Robert could work on the GC module not because he was considered the "best" person for the job but because he was the "right" person with the basic skills and the capability to learn fast enough to deliver quality code in time. As Robert broadened his skills to the new area, the team became less dependent on Dan for the GC module. Again, the group mode of coordination encouraged the versatility of members, resulting in less dependency on specific individuals and more flexibility in work distribution and work schedules.

LC members functioned as a group in external interactions too. The overlapping expertise alleviated the pressure for all members to be present in a meeting to represent

their areas of expertise. The same applied to customer support. Except when the issue required very specific expertise, members could cover for each other. For instance, in the email below responding to a potential customer, Dan suggests contacting Fred, who lived in the area, for further assistance:

> Date: 1998-7-23 04:22:16 GMT
> Subject: re: meeting with LC
> > Dan
> >
> > Please give me a call so that we can set up a meeting time. Thanks,
> > [name]
> > Ph xxx xxx-xxxx
>
> Sorry not to get back to you today (yesterday, on this coast).
> We're spread out across the country; for your area code, the
> right person to talk to is Fred (fred's email address), who works in Palo Alto.
> Is it ok if he calls you instead?
> Conference calls are also a possibility.

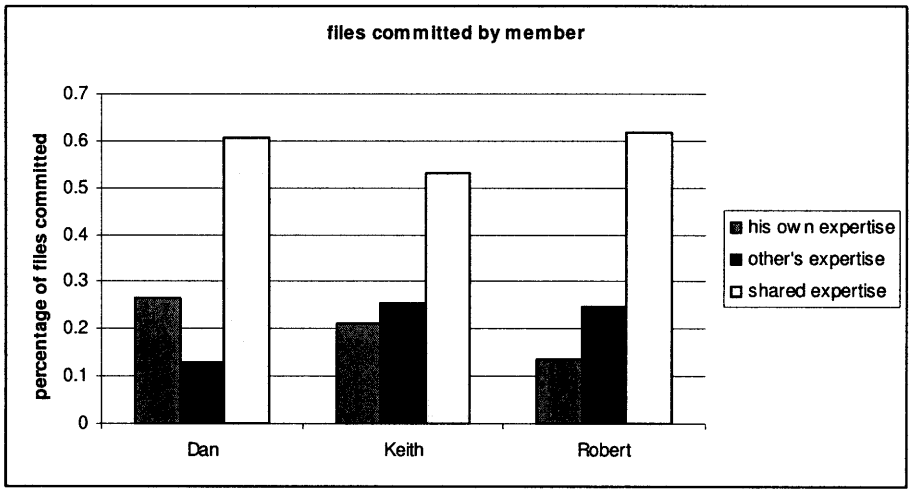
This broad base of shared expertise provided an added flexibility to the initial work distribution based on the pre-existing expertise of each member. [23] A preliminary analysis of the CVS log confirms that LC members actually worked on modules beyond their pre-existing areas of expertise. In Figure 4.9, the X-axis represents files committed by each member and Y-axis shows the proportions of committed files that fall into the area of each member's pre-existing expertise, the area of shared expertise, and the area of others' pre-existing expertise. Although the data is limited, [24] the result suggests a large

[23] Xu et al. (2005)'s study found that LC members distributed work based on multiple criteria, such as expertise, availability, and interest.

[24] The limitation comes from two main sources. First, the original CVS log does not include the information of who committed each file. This information was obtained by comparing each commit with update information in emails, and the author of 68.6% of total committed files was identified. Second, to identify the area of expertise that each file falls into, I used the expertise mapping conducted by Xu et al. (2005). They identified the expertise areas of each member from papers members had written and questionnaires administered to members, and then mapped them onto modules. Using this expertise mapping, I could identify 73.3% of total committed files for the location of expertise. As a result, slightly over half (52%) of the total files that have information on both were included in the analysis. Given the selectiveness of the data, the result is presented only to demonstrate the versatility of LC members.

proportion of tasks in LC were in the areas in which more than one of the members had proper expertise, allowing more flexible distribution of tasks. Also it confirms that members also worked on tasks that belonged to other member's specialty, allowing members to grow more versatile and less dependent on the expertise of one particular member.

Figure 4.9 Expertise of files committed by members



The broad base of shared expertise was critical for the group mode of coordination to be effective in LC, allowing flexibility without risking performance. As already pointed out, without this overlapping technical expertise, the time spent for group coordination can be substantial enough to diminish productivity. LC provides good examples on this point as well. LC experienced a prolonged procrastination in settling on a pricing model in spite of numerous customer requests for this information. In retrospect, Dan and Robert described the situation in an interview where all three full-time members were present:

> We would get email requests about things that were not technical issues. They were like pricing issues after we decided that our pricing model was wrong and we just got rid of it on the website.
>
> About eight months. We knew our pricing model was wrong but we didn't have a good one. So we just didn't say.
>
> And so we would get emails saying, 'How much does this cost?' And so the question is, 'who's going to respond to this,' you know. [LAUGHTER]

In another example, LC members worked extensively on developing a business plan between March and August 2000. They were, as Dan put it, "working very inefficiently and hating it, not doing a great job." In contrast to the technical tasks they were "incredibly" efficient in coordinating because they knew very well each other's expertise, it was never easy to coordinate tasks in which nobody had proper expertise but which still had to be done. In a group interview, the three full-time members were unequivocal in expressing the difficulty:

> Writing business model was something where we really did have to figure out who was writing what. Someone would write it and then someone else would tear it apart.
>
> Yeah. And after somebody would write it then somebody else would look at it and say, you know, this is junk.
>
> And so that went around....I mean, numerous times.
>
> It's just the amount of arguing and discord because none of us knew what the right answer was.
>
> We could see that we were generating wrong answers but none of us knew what the right answers were.

The group coordination that was very effective in coordinating technical tasks did not work well with tasks for which members did not have proper expertise. And these

tasks comprised everything in the category of "non-technical":

> There's a whole category of things that we just were not prepared to deal with and therefore didn't want to take care of, that were just falling on the floor and some of which we would spend an enormous amount time in a meeting trying to get [it done].
>
> Do it slowly, badly, and inefficiently.
>
> Who's going to do what? [Laughter]
>
> And eventually we would do something that was probably no better than it falling on the floor. But there is a clear place where Ray made a big difference and I think that's a good thing because he really picked up this big hole. And we knew there was this hole; it was obvious to us because we needed somebody to take care of that stuff.

The group mode of coordination was still used for the "non-technical tasks," but in this case not because they were all knowledgeable but because they all lacked basic expertise. Despite the time they spent together arguing, the outcome was often far from satisfactory and accompanied by repeated delays.

For a long time, LC members attempted to "fill the hole" by doing what they were good at. With limited marketing skills and resources, reengineering the product for the interested party was more "doable" than creating a good business model that would attract the right buyers:

> The thing is that we are incredibly more efficient about technical stuff. We can reengineer a big piece of our system in an enormous amount less time than it took us to write this business plan.

In fact, LC spent a lot of time internationalizing their product for a foreign distributor they came in contact with in the summer of 1998, and these efforts continued

intermittently and unproductively until early 2000. Robert recalled this experience in an interview:

> We went as far as to do all of the work to allow the internationalization of all of the messages and so forth. It turned out that this didn't really go anywhere at the end. And had it not been for him, we would have never done the internationalization. [...] But it's an example of something where we were chasing something and we did it for that reason, but it didn't pan out. [...] So you can't really decide whether it was worth it because you don't know what the opportunity cost really was. It could be that you lost something big because you didn't do the big project that you never got around to.

LC members continued their efforts to grasp what was hard to do on their own by translating it into what they do best. With lack of market information and marketing skills to define a target market and build an attractive business model, they tried to contain the uncertainty by perfecting their product to accommodate various market requirements, for example, operating on multi-platforms. Ray, who was finally hired to lead LC's marketing efforts, hinted at this pattern in an interview, saying it was "their nature":

> They've been close several times. Then they go back into the cocoon. They run out of money, they go back in, they say, oh, we want to do this, this, this and this. Instead of releasing the product, they go back in and start the development process again.

Conclusion

LC members embarked on a distributed software development project based on strong confidence that they were capable of better coordination with high flexibility. Nevertheless, LC members faced various mutual knowledge problems and struggled with the challenges of maintaining awareness of the task and the team in a distributed situation.

Through a series of early conflicts, members realized that successful distributed work required more than a simple stretch of good collocated work. Learning from doing, LC members gradually developed useful coordination practices over time.

LC's coordination practices are characterized by interweaving the personal and group modes of coordination according to the tasks and situations. For tasks that needed the expertise of a particular individual and close collaboration with that person, they use the personal mode of coordination through dyadic communication. For tasks that required general expertise or that affected everyone in the team, they used the group mode of coordination through group emails or phone meetings. As the task situation changed, the switches between the two modes were frequent. They not only coupled types of communication with the coordination needed, but also maintained the balance across different coordination types to achieve both efficiency and awareness. Members also increased communication and coordination of all types in periods of high activity.

This dynamic coordination was possible partly because LC members had overlapping expertise as well as unique specialties. Each member's specialty allowed him to work autonomously on modularized tasks, adjusting his work schedules more flexibly to their local situations. At the same time, the broad base of overlapping expertise among members made the coordination a function of the group in general, not the role of particular individuals. The group mode of coordination could be dominant and effective in LC, bridging possible gaps of mutual knowledge that might have been caused by the personal mode of coordination, because members were capable of understanding the task situation in other locations without paying too much time and learning cost. The group mode of coordination, then, facilitated flexibility by allowing dynamic distribution of

tasks beyond the preplanned division of work. A member could pick up a task and fill the hole in the work flow, thus minimizing idle waiting time to synchronize interrelated tasks. Moreover, it allowed members to broaden their skills into areas beyond the pre-existing expertise (as shown in the example where Robert became the linker expert of the team). As members not only deepened their skills in their own areas but also became more versatile, the group mode of coordination became more effective, which further ensured flexibility of when members chose to work on what. However, the same group mode of coordination did not incur this positive cycle in the case of "non-technical" tasks where members lacked basic expertise and experience. LC's group coordination practices highlight that temporal structures and norms in organizations are closely related to the way work is coordinated (Perlow, 2001). But they also indicates that the influence is not always straightforward, moderated by other factors, such as expertise.

LC's coordination practices demonstrate that coordination is not predetermined but "emerges from the interactions among the members of the team" (Hutchins, 1990, p. 207). It was particularly so in LC, where tasks were highly interdependent yet somewhat unpredictable, and thus hard to plan ahead. When work itself was emergent, so was its communication and coordination. As Dan put it, "You would like to think that things would be more planned than that, but people sort of went and did what they saw as necessary." For this ad-hoc coordination, then, members' willingness and capability to share the information that becomes the context of others' work are critical. The adaptive evolution of communication and coordination practices found in LC highlights that communication and coordination are not static team processes but dynamic capabilities members learn and develop over time. The prior research that reports negative effects of

geographic dispersion on team communication and coordination based on short-term observation or snap-shot comparison is limited for this very reason.

As a virtual team without established formal structures, data from LC provided a longitudinal window to observe how members gradually discover effective communication practices to coordinate their dispersed activities in a particular type of work, software development. It has been noticed that the domains of work where virtual teams are widely adopted are mostly populated by professionals who can operate to a large extent locally and autonomously, which brought up the question of whether some types of work are more suitable for distributed coordination than others (Hinds and Kiesler, 2002). Software development appears to fit this category. Software teams are increasingly going virtual, often spanning boundaries across the globe. Modularization enables individual programmers to work autonomously for some time. Although it necessitates dependency coordination among modules assigned to different members, it reduces the need for synchronous coordination considerably. Thus, for teams whose tasks require a high level of synchronous coordination among a large number of people, coordination might be more challenging. Nevertheless, it does not automatically lead to the conclusion that some types of work are inherently more effective than others when distributed, as team effectiveness depends on the team's capability to structure and adapt their communication to support the type of coordination needed. Also, considering that LC, working in the late 1990s, utilized rather simple media (phone and email) in dynamic and effective ways to coordinate various work situations, the increased choice of multimedia nowadays may give virtual teams the opportunity to better tailor their communication to meet their specific coordination needs. In short, more empirical studies

on a variety of virtual teams in different contexts—especially, type of work, membership, technologies in use—are called for to answer this question.

CHAPTER 5

Using genres and genre systems for temporal coordination

In the previous chapter, I showed how LC members coordinated their distributed activities, interweaving multiple media dynamically according to their task situations. In this chapter, I focus on how LC members used asynchronous emails—the main communication method of the team—to coordinate various tasks and activities over distance. The notion of genre and genre systems is used as analytical tool to examine email communication practices in LC. While LC members enacted multiple genres and genre systems in the course of the project, I focus on those whose socially recognized purposes in the team are more directly related to dealing with the problems of temporal coordination—e.g., scheduling meetings or events, synchronizing individual effort, allocating time to tasks (McGrath, 1990). The importance of temporal coordination in virtual teams has been recognized (Massey et al., 2003), but still little is known about how it actually occurs in practice, under what conditions, and with what consequences. This chapter attempts to answer this question by analyzing LC members' everyday email communication through the lens of genre and genre system.

Communication genres and genre systems

For studying communication practices in virtual work, the concept of communication genres offers a useful analytical lens. I draw on a structurational perspective (Giddens, 1984) to understand genre as a social structure constituted through

individuals' ongoing communicative practices (Miller, 1984; Orlikowski and Yates, 1994; Yates and Orlikowski, 1992). Central to this perspective is the recursive relationship between everyday practices and the social structures that are the medium and outcome of those practices. Social structures are constituted through recurrent human practices, which in turn are shaped by the structures.

Typical genres of communication in organizations—for example, memos, letters, reports, and meetings—are socially recognized types of communicative actions that are habitually enacted by organizational members over time to realize particular social purposes in recurrent situations (Yates and Orlikowski, 1992). Through such enactment, genres provide structuring templates that guide and shape members' communicative actions. Such ongoing genre use, in turn, reinforces those genres as distinctive and useful organizing structures (Orlikowski and Yates, 1994).

As organizing structures, genres shape beliefs and actions, and in doing so enable and constrain how organizational members engage in communication. Whether used explicitly or implicitly, genres powerfully influence the discursive norms of organizational interaction (Yates et al., 1999). One way of understanding these discursive genre norms is to examine the socially recognized or sanctioned expectations around key aspects of communication: purpose, content, participants, form, time, and location. Genres are "indicative of what communities do or do not do (purpose), what they do and do not value (content), what different roles members of the community may or may not play (participants), and the conditions (time, place, form) under which interactions should and should not occur" (Yates and Orlikowski, 2002, pp. 17-18).

As enacted social structures, genres may change over time (Yates and Orlikowski, 1992). They may be reinforced and stabilized as members routinely draw on existing genres to engage in communicative action. But genres can be altered or changed as members make modifications to existing genres or introduce entirely new genres, leading to changes in the group's discursive expectations. A genre lens thus offers a window onto the creation, reinforcement, and change of a group's communicative practices, and how it structures work temporally.

In order to capture a more complete picture of the interactions, however, the focus of analysis should be extended beyond individual genres to include the interdependent sets of genres routinely mobilized to structure work and communication. The notion of genre system—a series of interrelated communicative actions (Bazerman, 1994)—is useful for this purpose, because it focuses on how members use sequences of communicative actions to coordinate their activities over time and space (Yates and Orlikowski, 2001; 2002). For example, the genre system around job applications might consist of a firm's advertisement of a position, an applicant's letter and resume, a manager's call to arrange an interview with the person (or a rejection letter), one or more interviews, and a job offer (or a rejection letter). The genre system coordinates or choreographs the interaction between job applicant and hiring manager. Using a genre system lens enables us to view group interactions as paced and meaningful sequences of events, rather than as single, isolated occurrences, thus directing our attention to the very process of temporal and communicative structuring

In LC, three genres and two genre systems were found particularly useful in structuring members' distributed work over time. The coded data of the emails was used

to identify and examine the genres and genre systems recurrently enacted by the LC members. I used the coding scheme to construct genre definitions on the basis of the frequencies of various coding categories as well as common paper-based and electronic genres identified in previous genre studies (Orlikowski and Yates, 1994; Yates and Orlikowski, 2001; 2002). As some of the genres might be modified over time in response to task characteristics or experiences with the electronic medium, additional attention was paid to finding variant or new genres through an iterative analysis of message texts and coding categories. The coding categories used to define genres and genre systems for temporal coordination in LC are summarized in Table 5.1.

Table 5.1 Definition of genres and genre systems in LC

	Genres	Genre Definition
Task-specific genres	Status Report	Report the status or progress of work (Purpose=status report, Content= work related and technical, sender=LC member)
	Bug/Error Notification	Notify about a bug or an error members ran into conducing their tasks (Purpose=bug/error notification, Content= work related and technical, sender=LC member)
	Update Notification	Notify about an update on server (Purpose=update notification, Content= work related and technical, sender=LC member)
Phone meeting management genre system	Meeting logistics	Announce, propose or negotiate the date and time of a proposed meeting (Purpose=scheduling, Content=work related, sender=LC member)
	Meeting agenda	Propose or announce the purpose and content of a meeting (purpose=proposal, content=work related, sender=LC member)
	Meeting minutes	Document the proceedings, decisions, and action items in the meeting (purpose=report, content=work related, sender=LC member)
Group-authoring genre system	Circulated draft	Propose a draft (letter, document, etc) (purpose=proposal, content=work related, sender=LC member)
	Reaction to the draft	Provide comments or feedback to the initial draft (purpose= response or discussion, content=work related, sender= LC member)
	Revised draft	Propose a revised draft after integrating others' feedback (purpose=proposal or response, content=work related, sender=LC member)
	Final version	Circulate the final draft (often as a copy of outgoing email) (purpose=announcement, proposal, or discussion, content=work related, sender=LC member)

Table 5.2 Summary of data [25]

	Phase I	**Phase II**
Dates covering	3/1/1997 ~ 2/28/1998	4/1/1999 ~ 3/30/2000
N of emails	3474 (30.1%)	2948 (25.5%)
Main activity	Technical: Product development	Administrative: marketing
Technical content	83.2%	48.0%
Administrative content	18.5%	46.2%

The analysis in this chapter is based on a subset of LC's comprehensive electronic mail archive. Two particular twelve-month windows, referred to as Phase I and Phase II, have been selected for comparison. Phase I (March 1, 1997 ~ February 28, 1998) starts from when the company equipped its members with necessary infrastructure and ends around the *alpha* release of the product. It is this period during which the product's core system was mostly completed. Phase II (April 1, 1999 to March 30. 2000) starts around when their product was available to the public and ends when LC started to write their own business plan after a major merger deal went sour. During this period, the main efforts of the company were geared toward marketing and sales of the product (and the company). As the focus of the analysis is on the internal communication genres enacted by LC members, email messages sent by non-LC members (254 messages) were removed, leaving a subset of 6,422 email messages finally included in the analysis. Table 5.2 summarizes data in each phase.

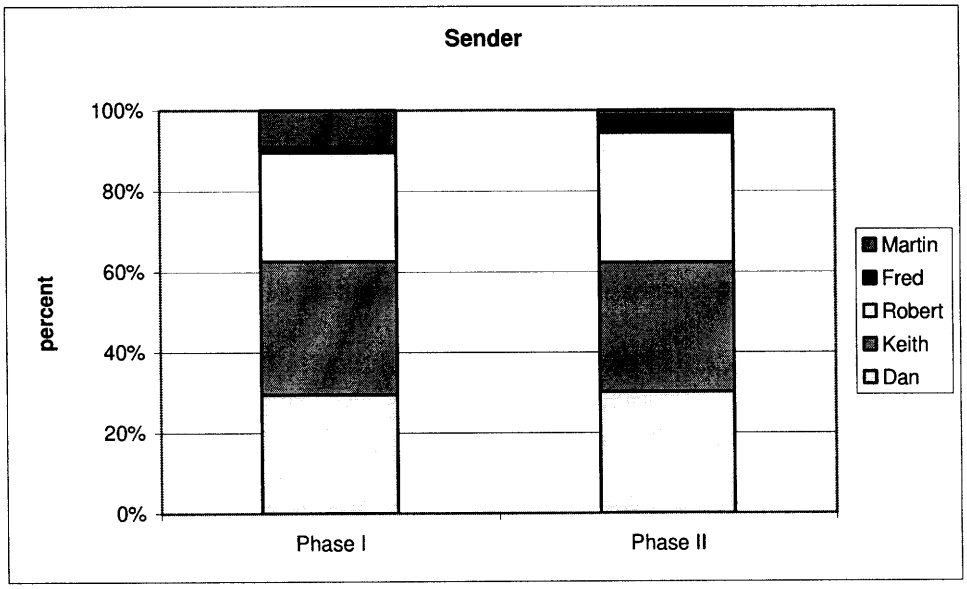
As indicated in Table 5.2, the topic of emails reflects the main activities of LC in each phase. In phase I, messages about technical aspects of the project outweighed those about administrative aspects, while emails regarding any aspects of work constituted

[25] The total in content may not add up to 100% because some messages included both technical and administrative content.

96.2% of total email communication. Although most messages were still about work (91.7%), administrative content (48.0%) slightly surpassed technical content (46.2%) in phase II, as the main development work had been completed and the focus of the company moved to marketing and sales of the product.

The distribution of messages sent by members during each phase is shown in Figure 5.1. In both phases, the three full-time members (Dan, Keith, and Robert) sent around 90% of total messages, with their contribution more or less evenly distributed. The contribution of the two part-time members (Fred and Martin) further dwindled in Phase II, as each left LC sometime before or during phase II.

Figure 5.1 Distribution of email messages by sender



In addition, the proportion of dyadic emails (those email exchanges between any two LC members) decreased in phase II (from 30.5% to 22.6%), as the need for dyadic collaboration on modules was reduced after the critical integration of parts had been finished. Group emails (those emails including at least the three full-time members as sender and recipients) increased in phase II (67.4% to 75.6%), as marketing, the area in which none of them was an expert, involved everyone.

In the following, I examine how LC members actually coordinate their work through enacting genres and genre systems in their email communication. Also I compare the two phases to track changes over time in the use of genre and genre systems in LC.

Genres used for Temporal Coordination

Of the many genres that LC members used in their distributed work, three genres were used explicitly for temporal coordination: status report, bug/error notification, and update notification. All three were related to technical tasks in building LC's system product (what are referred to as task-specific genres here), and thus were enacted mainly by the three full-time members.

Status report genre

In the status report genre, the sender reported the progress of his current tasks. A typical status report briefly summarizes the current status of the task (e.g., how particular tasks are progressing, how much time has already been spent on it and how much more will be needed to complete the task, what problems have been encountered, and what

kind of collaboration with other members might be needed to complete the task).

Consider the following example:

> Date: Sun, 23 Mar 1997 09:31:53 -0500 (EST)
> From: Keith
> To: LC
> Subject: status
>
> I just updated the server with attempts get the IntCode to zero
> errors. I have one more bug to fix (I am not doing the merge between
> a reference to an atom and a reference to an array properly.)
> There is only one place in all of the sources that we have that
> does this. LC has never written code that did this. In fact,
> Dan and I were unsure that this could ever arise in type correct
> code until we saw this case. It is not discussed in the specification at
> all. However, dan and I were able to assign a meaningful type
> assignment for this operation.
>
> I will fix this tonight or tomorrow.
>
> I would like robert to check Parser and TypeEncoder
> carefully when he does his next update.
> We were both hacking on these files at the same time. I think that cvs
> did the right thing since we were hacking different parts of these
> files. As I did updates, I saw changes to my copies that I did not do.
> However, a second check is not unreasonable until we get more
> comfortable with cvs merging.
>
> keith

This specific example of an early status report relates to all three of McGrath's (1990) mechanisms of temporal coordination—scheduling, synchronizing, and allocating. Reporting his progress with the bug fixing, Keith is scheduling his own work and allocating his temporal resources accordingly toward fixing another bug ("I will fix this tonight or tomorrow"). He is also trying to align his pace (synchronize) with that of another member ("I would like Robert to check ...carefully when he does his next update"), asking him to check some files at his next update in order to ensure that they work on the same version of the code.

In general, successful team coordination is based on members' awareness of tasks (e.g., what steps need to be taken next) and team (e.g., who knows what among the members) (Walther, 2002). To make distributed collaboration successful, members need to stay current with, for example, what other members are doing, who is communicating with whom, or what kinds of problems concern their tasks directly or tangentially. Status report messages provided such task awareness to LC members, and helped them manage the complex temporal dependencies between tasks assigned to different members. The following message—a request for status reports —demonstrates the importance of such information:

> Date: Mon, 10 Nov 1997 13:49:11 -0700
> From: Robert
> To: LC
> Subject: All quiet on the eastern front...
>
> Keith, Dan...
>
> Could you mail me an indication of when you expect to
> commit code for your current problems? I'm trying to
> make some plans for my time.
>
> ---
> robert

In order to assign tasks and solicit or offer help appropriately, members need to know each other's knowledge, skills, and motivations. LC was in a favorable position in that members shared similar technical, educational, and cultural backgrounds, and some of them even had prior experiences of collaboration. But virtual work requires ongoing alignment of members' perceptions about the roles and responsibilities of each member, the interdependencies among them, the current status of each member's assigned tasks, their availability for interaction, and so forth (Cramton, 2001). For instance, in the status

report message below, Robert asks for the approval of his update from the rest of the team to ensure that his plan does not interfere with others':

> Date: Fri, 24 Oct 1997 10:52:29 -0600
> From: Robert
> To: LC
> Subject: make stuff about done
>
> I've about got the make stuff ready...I'm just doing
> some final testing.
>
> I will soon be updating my code with the changes on
> the server. If you want to get anything into the inital
> porting version, please tell me and I'll wait a bit to
> do my update. Otherwise, I plan to update my world,
> build a link-only version of the front-end and
> link manager, feed it to the system, watch it crash,
> and then commit all of this stuff to the server.
>
> Let me know if this does not work with your plans...
> --
> robert

LC members enacted the status report genre to maintain ongoing awareness of their distributed tasks, individual progress, plans, and schedules. The frequent status reports helped members get the clear sense of "who's still busy on that problem" or to decide "whose court the ball is in now." Members used these status reports to determine which collaborative actions were required when, and how they should adjust their individual schedules, expectations, and plans accordingly.

Bug/error notification genre

LC members used the bug/error notification genre to report bugs or errors that they had encountered in the course of conducting their tasks. A typical bug or error notification has a subject line identifying the problem, and in the body, a detailed description of where and how the problem or error has occurred. As their geographic

dispersion precluded members from observing what was happening over each others' shoulders, any technical problems had to be documented properly so that each member could reproduce them on his local machine. Its importance is reflected in the email below, where Robert addresses the issue and proposes a few requirements for bug reporting:

> My problem is that I have not been able to reproduce the errors that Dan and Keith have been reporting. When I executed "make clean link5," it gave no errors. I did a make clean and tried to make everything, and now I'm stuck.
>
> I wish the bug reporting were more clear about what commands are needed to produce the error messages. A screen copy showing the directory, command entered, and error would be most helpful. Please edit out billions of lines of unrelated output.
>
> robert

Robert's suggestion was adopted and put into practice immediately. See the following example:

```
Date: Tue, 6 May 1997 16:33:03 -0400 (EDT)
From: Keith
To: LC
Subject: linker bugs
Robert,

N->Q
RO->Q
R->RO
LAny;->RO
V->Q
Link.InternalLinkError
        at Link.ObjectFile.addRelocationTranslation(ObjectFile.java:710)
        at Link.ObjectFile.processRelocations(ObjectFile.java:574)
        at Link.ObjectFile.processRelocations(ObjectFile.java:638)
        at Link.ObjectFile.processRelocations(ObjectFile.java:638)
        at Link.ObjectFile.processRelocations(ObjectFile.java:567)
        at Link.ObjectFile.processRelocations(ObjectFile.java:567)
        at Link.ObjectFile.processRelocations(ObjectFile.java:567)
        at Link.ObjectFile.processRelocations(ObjectFile.java:638)
        at Link.ObjectFile.processRelocations(ObjectFile.java:638)
        at Link.ObjectFile.link(ObjectFile.java:262)
        at Link.ObjectFile.link(ObjectFile.java:182)
        at Link.ObjectFile.linkDriver(ObjectFile.java:1359)
        at LC.Compile.main(Compile.java:136)
make: *** [link5] Error 1
/lc/src/boot/Tests(19)
Here is what you have to do to recreate this bug.
```

1) cvs update
2) cd whereever/LC/src/boot/
3) make clean all
4) if you did not install sun's java at /java,
add a symbol in your environment called SUNJAVA that points to where
you did install it.
5) cd whereever/LC/src/boot/Platforms/xscale/LClib
6) type make
7) cd whereever/LC/src/boot/lib
8) type make
9) cd whereever/LC/src/boot/Tests
10) make link5

it should fail just like the above.

sorry for the complicated procedure, we are a little behind on getting
the makefiles into shape as we have started linking with real
libraries. I will be working on this while you hack the linker.

I will be leaving for hiking at 5:30edt, so if you have troubles
recreating this, get me soon.

keith

As requested by Robert, Keith included the screen copy of the error, showing where and how the error occurred and step-by-step instructions of how to reproduce the bug. This soon became established as the standard format of bug/error notification genre. Furthermore, members took extra care to specify any possible discrepancies between their local computing environments. In the same example, Keith added an additional instruction (step 4) to remind Robert of a possible difference in their machine directory structures.

Bugs or errors have unpredictability as their primary temporal characteristic. They could be quite disruptive to LC's work. Their occurrence meant that a new task (fixing the bug or error) had to be allocated to someone and that person had to adjust his work schedule to accommodate this unexpected demand. Often, members working on related modules had to suspend their current tasks until the problem was solved. In short, a bug or error notification message signaled the necessary readjustment of members' task

schedules. In his interview, Robert vividly described how these contingencies affected his daily work schedule:

> The first thing I do in the morning is just kind of look at the massive quantities of email that we get and I have to sort through and just kind of decide the thing that I have to do deal with right now. Are there any disasters going on, any new stuff that, you know, somebody found a bug that is something that I should be fixing? And so, I kind of go through that and decide whether there's something there that's going to really change what I thought I was doing today. And then, after I've taken care of that, then it's kind of back to remembering, OK, now what was I going to do today?

In most cases, LC members had little difficulty assigning these emerging tasks among themselves:

> There's lots of just finding a bug when we are just sitting there and it's not territorial but we all know our expertise. We know 'who is the real expert on any kind of a thing', and we've gotten good enough so that we can kind of smell a bug. You know, just put your hands down on the computer and close your eyes and the Force will tell you [laughter] — it's Dan's, Robert's or my bug and we get it right most of the time.

As Keith's comment above illustrates, members often instantly knew to whom to pass the bugs or errors based on their understanding of each other's technical expertise (or their ability to "smell" whose bug it was). This tentative decision was explicitly communicated through the bug/error notification genre, and then confirmed via the ensuing email discussion. Often, LC members used the subject line to toss the problem to one another, as shown in the following examples:

> Subject: The next bug has Keith's name on it.
> Subject: I think that the ball is now in Robert's court
> Subject: live problem has Dan's name all over it.

Typically, a bug or error notification indicated that the sender needed assistance from other members. Because a delay in fixing bugs or errors could cause long interruption of other tasks, such requests were considered as high priority. A bug or error notification message punctuated the rhythms of individual members' daily schedules, requiring them to dynamically readjust and recalibrate their tasks and time, on a daily or even hourly basis.

Update notification genre

LC members used the update notification genre to report the completion of a task and to indicate that an update to the LC system follows. A typical update notification has a distinctive subject line of "new [type of update, e.g., file, code] on [location, e.g., server, web]," followed by a short description of the update in the message body. In the example below, Robert notifies other LC members that he has uploaded a new code file to the server that fixes some previously identified errors:

> Date: Thu, 13 Nov 1997 13:01:59 -0700
> From: Robert
> To: LC
> Subject: new code on server
>
> I just checked in changes to the dependency
> generation/tracking code that fixes the
> problems with too many dependencies in the
> database.
> --
> robert

In the early months of LC project, update notification was generally a part of the status report, which included a list of other tasks, both completed and still in progress.[26]

[26] See the first example of the status report on page 154, where Keith begins his message with "I just updated the server with ..."

Over time, members started distinguishing the update notification as a separate genre to keep track of the latest system updates. In his interview, Dan recalled:

> We just tried to be, I think, a little more stylized. ... And so, there's this whole big folder of the "new on server" messages. And I don't remember when we started that, but we said that it was really important for purposes of just being able to figure out when something happened. Because you need to know when you go to make a release, you need to know what you did.

As system updates typically affected everyone's work, the distinctive subject line ("new...on server") functioned as an alert, highlighting relevant temporal information. With this common format, members could easily identify the update notification messages. Later, the update notification genre was further standardized in its form, as illustrated below:

> Date: Tue, 21 Sep 1999 10:36:07 -0400 (EDT)
> From: Kieth
> To: Dan, Robert, Martin, Fred
> Subject: new code on server
>
> Also fixed a couple of serialization problems where we were not compatible with some of ms's undocumented data structures.
>
> This also fixes some bugs where sublists of array were not properly managed.
>
> Also fixed a bug where AbstractTransform did not throw the documented exceptions.
>
> keith
>
> Added AbstractTransform and it's associated stuff.
>
> CVS: ----------
> CVS: Enter Log. Lines beginning with `CVS: ' are removed automatically
> CVS: Committing in .
> CVS: Modified Files:
> CVS: [11 files were modified]
> CVS: Added Files:
> CVS: [4 files were added]
> CVS: ----------

In addition to the subject line format, the CVS log was included at the end of message body. This new feature relieved members from typing in the specific location and details of changes, precluding any errors and semi-automating the update notification.

Update notification messages were sent approximately at the same time as the actual updates on the LC system. This temporal proximity of action and communication about the action is indicated by the various temporal references in update notification messages, for example, "now," "just," "I am about to." The update notification genre thus functioned as a synchronization mechanism, providing the "virtual temporal symmetry" (Orlikowski and Yates, 2002). Weekly phone conferences allowed the members to achieve temporal symmetry for a short time, but members also created the effect of temporal symmetry through email. The temporal proximity between the actual system update and its notification increased temporal symmetry by guiding members to the most up-to-date code or object. Together with the synchronous phone meetings, the virtual temporal symmetry created by use of the update notification genre helped LC members maintain adequate temporal coordination to support their distributed software development.

Interweaving genres in practice

The three genres were often interconnected, structuring communication among members. As an example, I have selected one work day on which to analyze the use of genres in actual practice. Table 5.3 and Figure 5.2 summarize the results. On this day, four members (Dan, Keith, Robert, and Martin) sent a total of 13 group email messages, which comprise two communication episodes, defined as a unit of temporally and

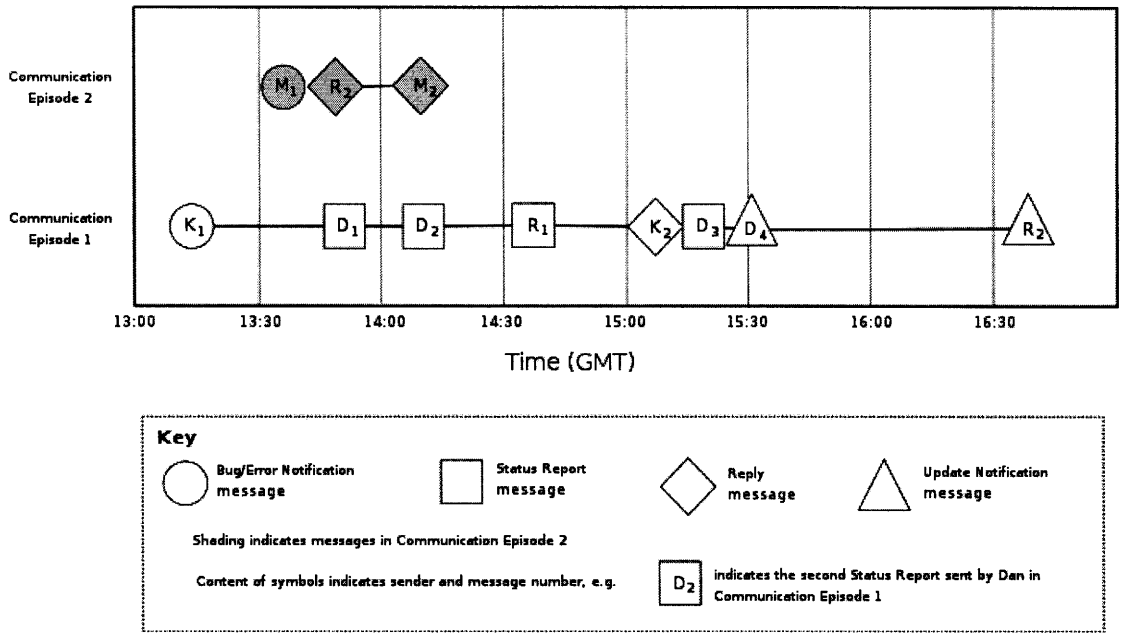
topically bounded communication activities. In these two communication episodes, the three task-specific genres were interspersed with related reply messages.

Episode 1 (non-italic rows in Table 5.3) was initiated by Keith's bug/error notification, describing the problems with the GC function of the system. His email message indicated that he had also been discussing these problems with Dan on the phone ("Dan has a grand theory of what is going on. He thinks it is GC related and is formulating mail even as we speak"). It was through the email, however, that the problems were officially announced and communicated to the rest of the team. About an hour later, Dan sent around a status report detailing his "grand theory" of the likely GC bugs and an approach for the fix. Dan and Robert then divided up the task and continued to "hack" them separately, while staying informed about each other's progress through ensuing status reports. Finally, Dan and Robert sent two separate update notifications announcing that new fixed code had been updated to the server. This communication episode took roughly 3 hours 20 minutes from the initial notification of the problems to the final announcement of the updated system.

Table 5.3 Genres enacted by LC members on a work day

Subject line	From	Episode	Genre
gc problems	Keith	Episode 1	Bug/error notification
Problems compiling (any suggestions)	*Martin*	*Episode 2*	*Bug/error notification*
Re: Problems compiling (any suggestions)	*Robert*	*Episode 2*	*Reply*
Today's GC bugs	Dan	Episode 1	Status report
Thanks Re: Problems compiling (any suggestions)	*Martin*	*Episode 2*	*Reply*
Confirmation of bug hypothesis	Dan	Episode 1	Status report
Re: Confirmation of bug hypothesis	Robert	Episode 1	Status report
Re: Confirmation of bug hypothesis	Keith	Episode 1	Reply
Re: Confirmation of bug hypothesis	Dan	Episode 1	Status report
Stuff checked in.	Dan	Episode 1	Update notification
More stuff checked in	Robert	Episode 1	Update notification

Figure 5.2 Timing and sequence of genres enacted by LC members on a work day



Episode 2 (italic rows in Table 5.3) is much simpler. Martin, one of the two part-time members of LC, initiated the episode by reporting compiling problems. Robert immediately responded with solutions since the problem had already been known to some of the LC members including himself ("I sent mail when Dan had this problem and we talked about it on Wednesday"). Apparently, Martin missed the weekly phone conversation when the problem had been discussed. This communication episode, which lasted only 25 minutes, concluded with a short "thank-you" note from Martin.

As shown in these two communication episodes, it was the norm rather than exception in LC that emergent problems required adjustments of broadly defined individual work schedules. The way a day unfolded often took a form of opportunistic structuring (Orlikowski and Yates, 2002; Yates and Orlikowski, 2001; Yates et al., 2001),

where members chose to provisionally halt the current tasks so as to concentrate on the emergent tasks. Coordinating these exigencies by enacting a series of task-specific genres shaped individual members' "loosely structured day" into a concrete form. Also it was through the enactment of these genres that new tasks were announced, assigned, worked through, and closed. The distributed collaboration of LC members was structured through members' ongoing and interconnected enactment of these task-specific genres.

Using genres over time: comparison between two phases

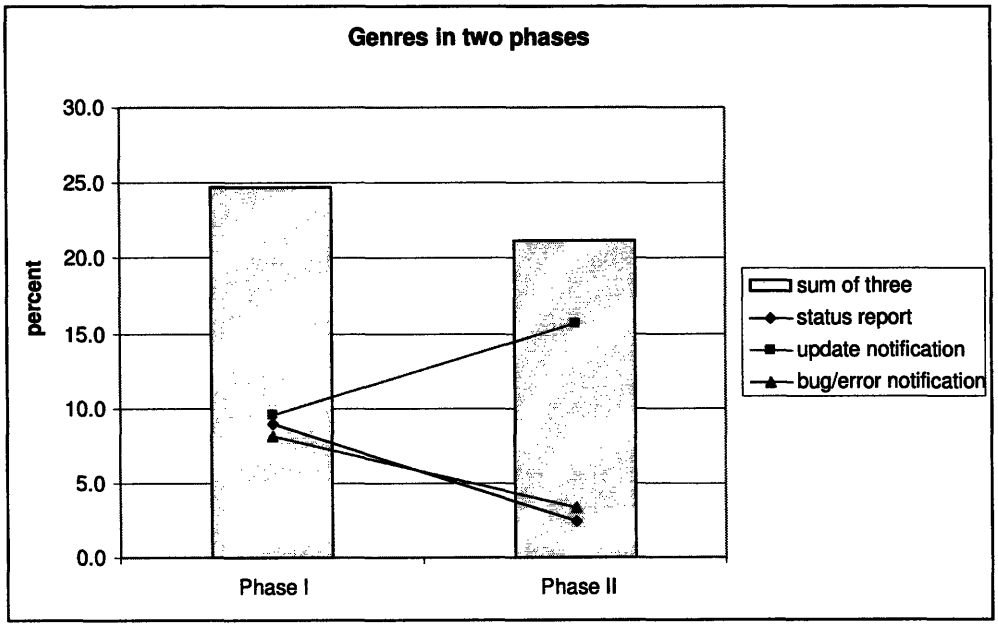
Change in use pattern: fluid with task situations and contextual changes

Reflecting the change in main tasks, from technical development in phase I to marketing in phase II, the significance of the three genres in terms of their proportion in the total messages decreased in phase II (see Figure 5.3). At first, the change seems marginal, a slight drop from 24.7% in phase I to 21.2% in phase II. However, it becomes more noticeable when each genre is taken individually for comparison. In phase I, the three genres occupied almost equivalent proportions (9.0%, 9.5%, and 8.2% in the order of status report, update notification, bug/error notification). In phase II, the status report and bug/error notification decreased to only a fraction (2.5% and 3.3%) of the total messages whereas update notification messages (15.6%) increased.

The changes in the make up of LC's "genre repertoire" (Orlikowski and Yates, 1994) across the two phases reflect not only different task environments but also the evolving relationships between genres. First, in phase II when the main developmental work had been completed with the product available on-line for public downloading, the

status report and bug/error notification were less needed. In particular, most serious bugs had already been addressed in the preceding years, and the incident of introducing new bugs by LC members became far less frequent, therefore decreasing internal bug/error notifications.[27] Second, the decrease of the status reports in phase II is closely related to the increase in update notifications. Besides the reduction of technical tasks that take long hours and more interim status reports in phase II, the frequent update notifications seem to have incorporated the role of status reports.

Figure 5.3 Genres in two phases



[27] LC also had a separate mechanism to have users report the bugs. Once a user filled out the required fields and submitted the form on LC's website, it was delivered to the email boxes of each member.

Figure 5.4 Genre use by three full-time members

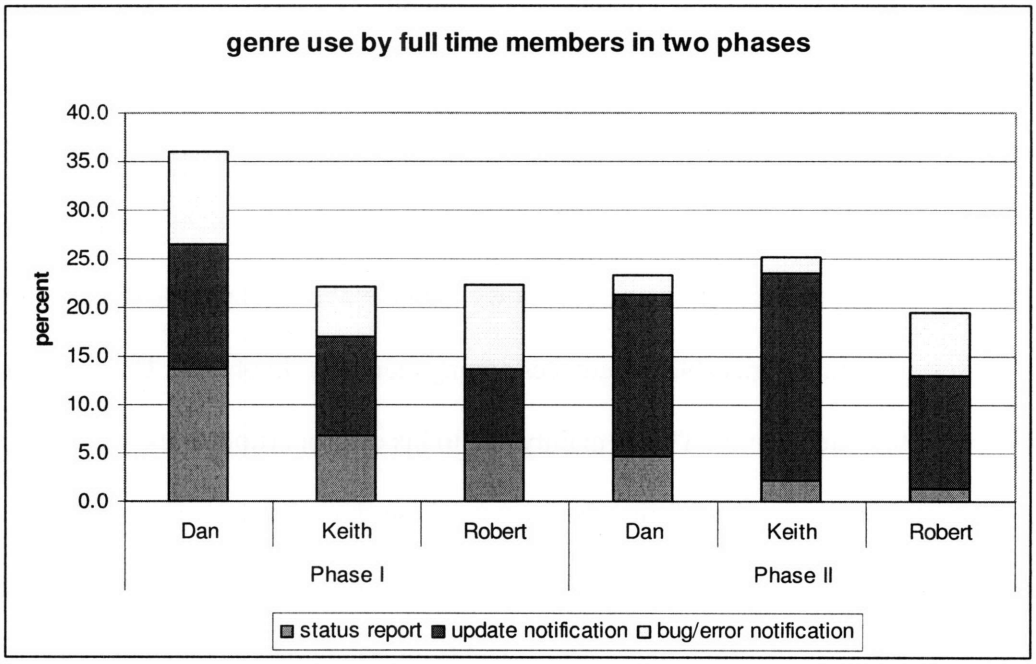
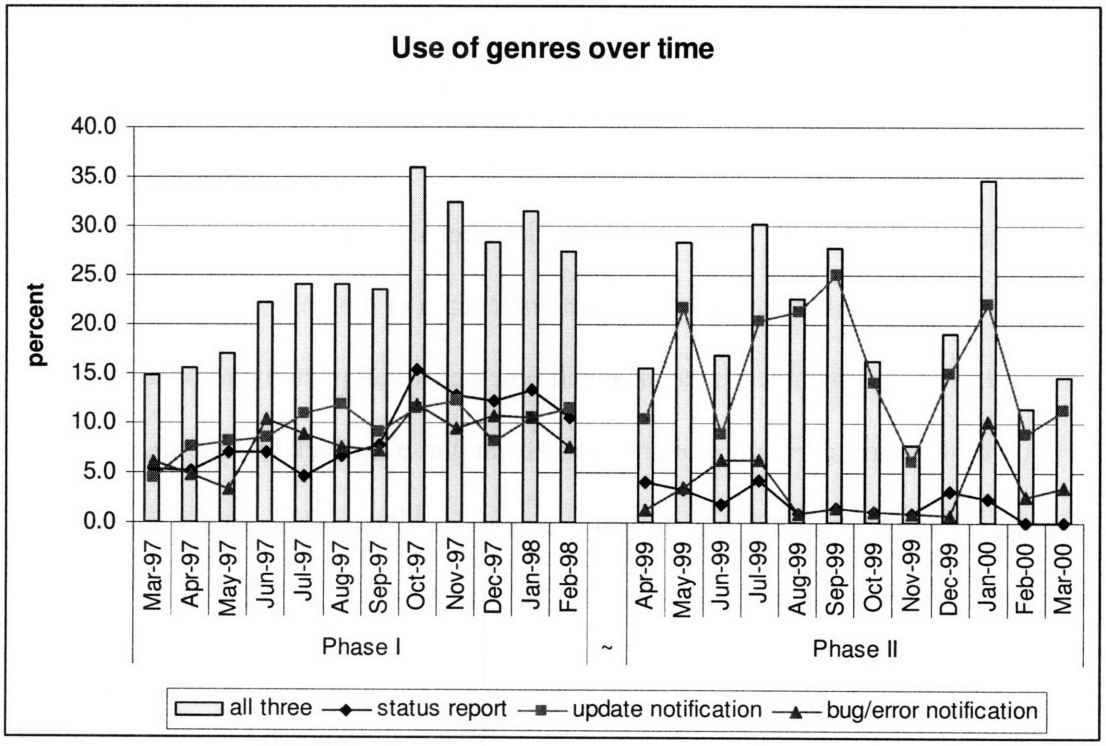


Figure 5.5 The use of three genres in LC over time



Given their technical subject, the three genres were enacted mostly by the three full-time members. However, as I pointed out in the previous chapter, LC had a norm of sending these messages to all of the team, including the part-time members who were not directly involved in the technical work. The members clearly recognized these genres as a means of enhancing the members' awareness of tasks and the team, therefore facilitating coordination at the team level. Figure 5.4 shows each member's use of the three genres in two phases, revealing noticeable differences among members. In phase I, Dan stands out in his use of the three genres. This is mainly due to his disproportionate use of status report, more than double that of the other two. In phase II, each member's use of the three genres became more or less evenly distributed (23.4%, 24.7%, and 19.2% for Dan, Keith, and Robert), since by that time, these genres had been stabilized and the overall use of status reports had been dropped.

The use pattern over the two 12-month periods shows the change in each genre's significance according to the project cycle (Figure 5.5). The use of the three genres by the full time members increased steadily in phase I. The three genres, which accounted for only 14.9% of total message volume in March 1997, increased to about 30% by the end of the phase I. This general increase in phase I reflected the first year activities geared toward the delivery of the prototype. In the first couple of months, which was the initial stage of the project, members spent as much time on overall planning as on actual code writing. By contrast, during the last few months in phase I, which was the final stage of prototype development (and they missed a release deadline initially set as October), members' efforts were concentrated on integrating and testing the system. Accordingly, members enacted the status report more frequently, also with high use of update

notification and bug/error notification. Overall, the three genres became more prevalent in LC's genre repertoire, as their software development activities advanced (thus generating more bugs, updates, and code releases) and as they became more familiar with the genre norms.

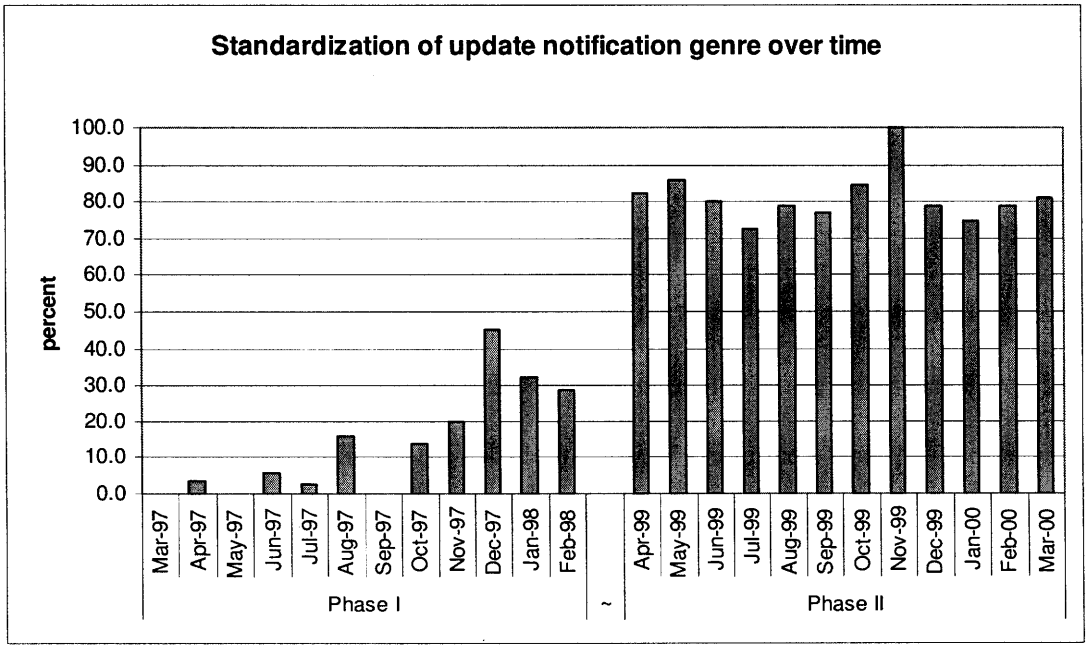
In contrast, the overall use of the three genres in phase II shows more fluctuations. LC had more frequent public releases with less serious modifications, interspersed with primary marketing activities (e.g., preparing press releases, visits to potential customers, writing business plans), and the use of the three technical genres punctuated with these rhythms.. The lowest point of November 1999 corresponds to the month during which LC members were busy preparing for a meeting with a potential merger partner. After January 2000 when LC had another release, the proportion of the three genres in LC emails was reduced to less than 15%, around where it was three years earlier. When the genres are viewed individually, the demise of status report is apparent as its proportion nears zero toward the end of phase II. The bug/error notification genre was also marginalized, except for the months subsequent to the new releases. Overall, the three genres became less prominent in LC's email communication, as the company entered into another stage of product development, and some of them became rarely used.

Change in form: providing fixity in communication

A genre has a socially recognized form and purpose within a community. These genre norms may not clear at the beginning but become stabilized as members repeatedly enact them in recurrent situations. Also genres can be altered as members make modifications to existing ones or introduce entirely new ones. The evolution of update

notification genre is a good example to show how a genre emerges and establishes itself with a distinctive purpose and form. In fact, this particular genre became highly standardized, enough to be recognized by glancing at its form. As already mentioned, update notification had branched out from the status report, as members felt the need to communicate the exact time and content of updates. Out of habit, members often mixed status report and update notification, but over time, update notification took a highly standardized form (first with its unique subject line, and later with the inclusion of the CVS log) and became the most discernable genre in LC. The Figure 5.6 shows the standardization of update notification over time in each phase: the first standardized form (with a subject line of "new --- on server") appeared as early as in April 97, but it took some time for this form to be stabilized. However, in phase II, this standardized form became a defining element of the genre.

Figure 5.6 Standardization of update notification in its form



As LC members became familiar with genre norms, they learned to enact each genre more precisely (single genre message) rather than blending several genres in one single email (hybrid genre message). I examined the proportion of single and hybrid genre messages where members enacted at least one of the three genres. LC members used each genre quite precisely even in phase I, with only 8.1% hybrid messages, but the proportion of hybrid messages further decreased to 1.1% in phase II.[28] The decrease of status report itself in phase II contributed to the overall decrease of hybrid messages, since the hybrid of status report and update notification accounted for over the half (67%) of all hybrid messages. Still, it is clear that members enacted for more single genres than hybrids in phase II.

Genre systems for temporal coordination

Temporal coordination of work typically involves a series of interactions through which different temporal structures and interests of individual members are expressed, contested, and negotiated. Among these several distinct sequences of communicative actions routinely enacted by LC members, two genre systems were particularly relevant to LC's coordination: the phone meeting management genre system and the group authoring genre system.

[28] The hybrids of status report and update were the majority (51 messages), compared to status report and bug/error notification hybrids (17 messages) and bug/error notification and update notification hybrids (7 messages). There was only single message that blended all three genres. By genres, bug/error notification was enacted most exclusively in a single genre message (93.5%), followed by update notification (92.6%), and then status report (82.1%).

Phone meeting management genre system

Scheduling meetings or events is one of the mundane but essential tasks of organizations. Given the dispersed structure and varied individual schedules, creating opportunities for synchronous interaction in LC required some coordination of its own.

The phone meeting management genre system was developed to coordinate phone meetings. It consisted of three interrelated and interlocking genres that preceded and followed actual phone meetings: meeting logistics, meeting agenda, and meeting minutes. [29] Not all occurrences of the phone meeting genre system involved the use of all three genres. Meeting minutes appeared very rarely, whereas meeting logistic was enacted more frequently.

The first genre, *meeting logistics*, was typically used to communicate the date and time of the proposed phone meeting, and (sometimes) alternate phone numbers to use for the meeting. The meeting logistics genre included messages not only proposing or announcing a meeting but also clarifying or trying to change the planned meeting time.

The *meeting agenda* genre was used to propose or announce the purpose and content of the meeting. In many cases, the meeting logistics and meeting agenda genres were enacted in a single message, as seen here:

> Date: Tue, 01 Apr 1997 15:08:33 –0700
> From: Robert
> To: LC <all@LC.com>
> Subject: Technical discussions
>
> Fred and I would like to start having regular technical discussions. Fred proposed having the first one on thursday morning. Please send me a list of times that fit your schedule and any items for the agenda.

[29] Although the meeting itself is also a communication genre, it was excluded for two reasons. First, the focus of the analysis in this chapter is on email genres, and second, phone meetings were not documented enough to conduct a genre analysis.

Proposed agenda:
0) schedule for technical discussions. Every week? Every other week?
1) Strategy and timetable for bolting together the parts that we've almost finished.
2) writing white papers on our technology
3) general Q and A on current activities.
---
robert

These two email genres that preceded the actual phone meeting often included a long thread of messages, each constituent genre appearing repeatedly. These threads of emails offer interesting examples for observing the dynamic and negotiated nature of temporal coordination in LC. Let's follow further the email exchanges ensued by Robert's email above. It is actually the first phone meeting genre system initiated after they decided to keep a weekly meeting schedule. In the email, Robert initially proposed Thursday, but it was finally announced to be Friday morning, mainly because Dan was not available on Thursday due to a dental appointment. Still, the meeting eventually occurred without Fred, who was attending an exhibition at that time.

In scheduling their second meeting, members attempted to regularize meeting schedule by holding it on the same day as the previous week. Unfortunately, Robert found he had a schedule conflict just one day before the meeting day and asked for a change, initiating the next round of the meeting management system:

Wed, 09 Apr 1997 05:04:38 -0600
From: Robert
To: LC <all@LC.com>
Subject: Technical meeting

Much to my chagrin, I've got a schedule problem with the technical meeting at 9:30pdt, 10:30mdt, 12:30edt on Friday. Which of the following options is best for you?
1) Do it at the scheduled time without me.
2) Do it friday afternoon (send times).

3) Do it thurday morning (send times).
4) Do it Thursday afternoon (send times)
5) Skip it this week.

Wed, 09 Apr 1997 10:09:21 -0400
From: Dan
To: LC <all@LC.com>
Subject: Re: Technical meeting

At 05:04 AM 4/9/97 –0600, Robert wrote:
>1) Do it at the scheduled time without me.
Or me, since I am probably on my way back from 1.5 hrs at the dentist,
With a jaw full of novocain and probably not in the best moods
(I'm having a crown replaced).
>2) Do it friday afternoon (send times).
2:30-4:30 works for me.
>3) Do it thurday morning (send times).
>
>4) Do it Thursday afternoon (send times)
Thursday is bad, especially this Thursday.
>5) Skip it this week.
That fits my schedule, too.

dan

It was not easy to reschedule the meeting with the short notice. Dan's dental appointments further limited options. It was finally rescheduled but went without Dan who had to withdraw at the last moment due to pain from his dental treatment.

The early examples of the genre management system highlight the difficulty members experienced at the beginning of their collaboration in finding a time good for all members. However, it was also through this early experience that members established some important genre norms. For example, members learned to specify the meeting time in three different time zones, after experiencing some confusion as evidenced in the email excerpt below:

> 2:30-4:30 works for me.
is this edt? mdt? or pdt?
is this carved in stone?

With some trials and errors, the phone meeting gradually emerged as a regular weekly event within LC, associated with a particular calendar/clock time (e.g., Wednesday 3pm EDT/1pm MDT/12pm PDT). Once this new temporal regularity was stabilized, LC members simply referred to it as "regular time" or "standard time," when they tried to schedule additional phone meetings. In the following example of email sent later in their first year, the "regular time" is mentioned as a point of reference:

> Subject: meeting Thursday
>
> Keith asked me to send mail asking for a meeting tomorrow (thursday).
> He spoke with Gus today and would like to talk about it.
> He suggested the regular time. If this does not work for you, send
> some alternate times and we'll try to arrive at a common one via email.
> ----
> robert

The third genre, the *meeting minutes* was rarely circulated through email. From the very beginning, Robert kept notes of every phone meeting; these notes were handwritten, very brief (typically one page long) and informal. Members knew Robert was keeping records about the meetings, and Robert occasionally reproduced them in electronic form to share. [30] The email below suggests that later Robert opted to put the minutes on the server:

> Date: thu, 04, Dec 1997 16:24:05 -0700
> From: Robert
> To: LC
> Subject: telephone meeting minutes
>
> I just added a directory [name] along with
> a word document [title].rtf. I'll be glad to
> produce other meeting dates upon demand---just send
> me email with the date, or request them in the meeting.
> --
> robert

[30] The first meeting minutes Robert circulated by email goes back to January 1997.

Such requests for minutes from other members were not frequent. However, when there arose some confusion about any decisions made during the meeting, Robert's minutes provided a good reference. In the email below, for example, Robert confirms the meeting date based on the minutes:

> Subject: re: conf. call
>
> Fred wrote:
>
> > Are we having a conference call today? or is my memory of when the next
> > call was to happen faulty?
>
> According to the minutes from last week, we agreed to hold the next
> meeting Thursday June 3rd at the usual time.

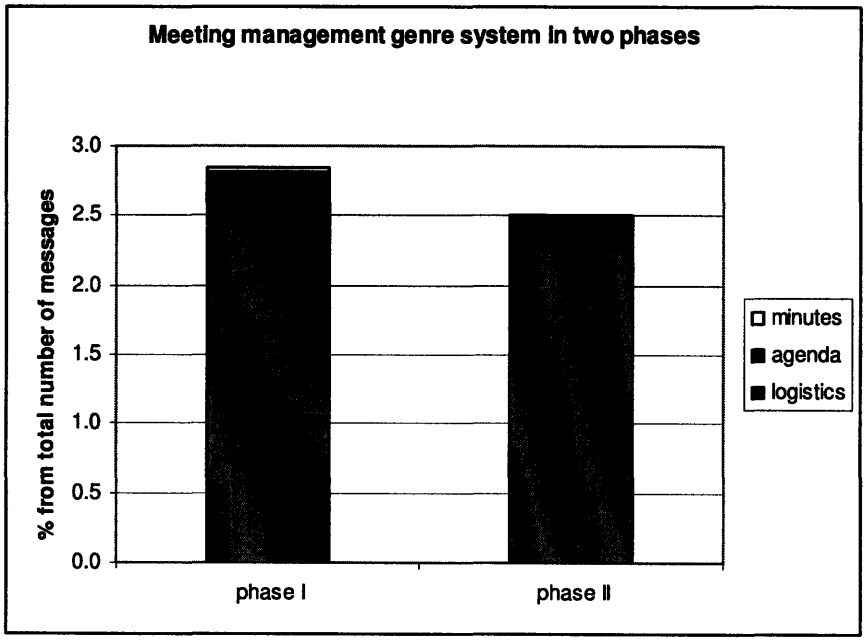
In another example below, Robert sent around the meeting minutes to the rest of the team to remind them of work assignments that had been decided in the meeting. Martin, who had been absent, appreciated it very much and replied:

> Subject: re: minutes Wednesday 16 September 1998
>
> Robert wrote:
> >
> > LC
> > minutes Wednesday 16 September 1998
> >
> > absent:
> > Martin
> > Things to do before beta:
> > - Must do
> > user_guide.txt Robert -> Martin
> > release notes Dan, Keith
> > license expire code Fred -> Robert
> > [...]
> > - Should do
> > [...] Keith
> > - Put off for later
> > [...] Robert
> >
> > Beta release delayed by two weeks (until 30 September)
> > next meeting Wednesday 23 September, regular time
>
> Thank-you Robert, it was a good idea for all of us to see the minutes of
> this meeting. I wish I had thought to suggest it; luckily you did.

Comparison between two phases

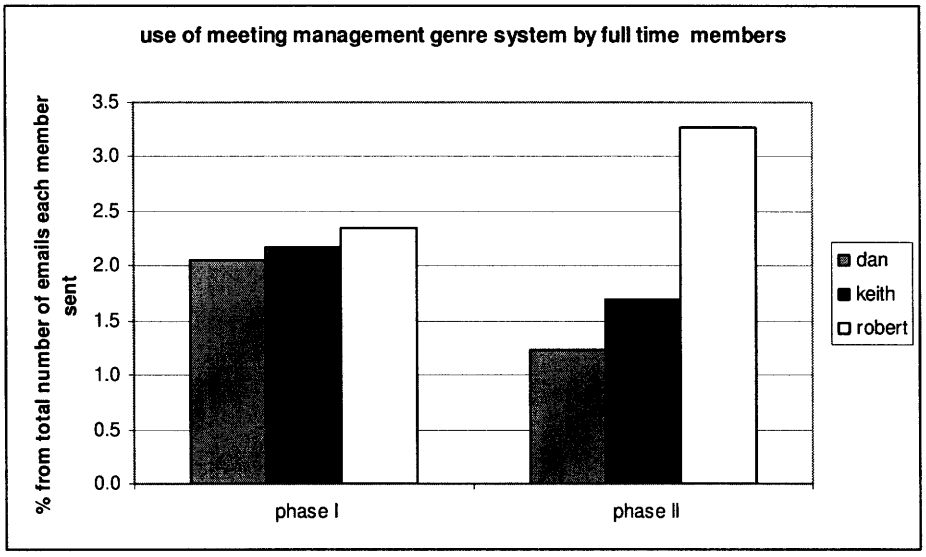
The meeting management genre system slightly decreased from phase I to phase II (see Figure 5.7). I identified 31 cases of meeting management genre system with 92 emails in total (2.7%) in phase I and 24 cases with 69 emails (2.3%) in phase II. Although its relative proportion in LC's genre repertoire remained similar in the two phases, LC members enacted fewer cases of the meeting management genre system in phase II. All LC members, including two part-time members, enacted the meeting management genre system. Figure 5.8 shows the three full-time members' genre use, reported as the proportion of this genre system in their email. In phase I, the meeting management genre system emails accounted for a similar proportion of each full time member's email, but in phase II, its proportion in Robert's email increased.

Figure 5.7 Messages in meeting management genre system in two phases[31]



[31] As a single e-mail message can be coded for more than one genre, the sum of the three genres shown here is slightly bigger than the total number of messages in meeting management genre system.

Figure 5.8 Use of meeting management genre system by full-time members



A close reading of emails suggests a reason for this discrepancy. Robert was often the only and last person who was notified of the possible rescheduling as shown in the following example:

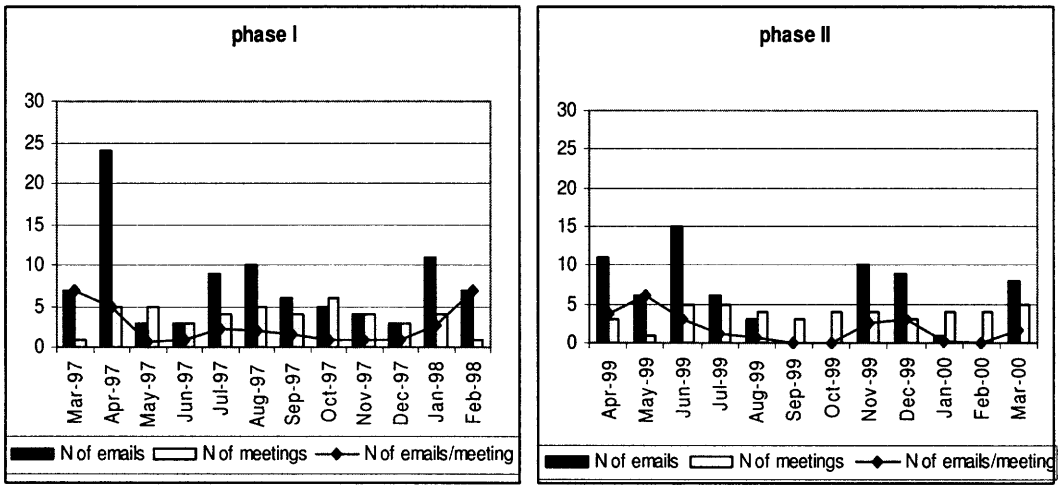
> Keith wrote:
>
> > Dan and Fred have said that 9:30pm est 6:30pm pst [is fine]
>
> This is fine with me.
>
> robert

Naturally, Robert was in the position of confirming the final meeting schedule and generally took the role of coordinator, sending out a reminder. As the role of part time members diminished in phase II due to their limited involvement in LC, Robert's participation became more noticeable in phase II.

When the number of phone meetings that are actually held is taken into account, the decreasing use of the phone meeting genre system becomes more evident. If the

frequency of messages enacting the meeting management genre system is directly proportional to the number of meetings, then messages in the meeting management genre system should decrease only if members had fewer meetings in phase II. However, according to Robert's meeting minutes, LC had same number of meetings (45 meetings) in each 12-month period. Figure 5.9 shows the number of emails in the meeting management genre system (red bar graph), number of meetings actually held (yellow bar graph), and the ratio between the two (number of emails divided by number of meetings, line graph) by month. There seems no big difference between the two phases in terms of the number of meetings held monthly, but the number of emails had clearly decreased in phase II. In particular, a few months in phase II (September and October 1999 and February 2000) show no trace of emails enacting the phone meeting management system, although the number of meetings actually held during these months remained close to the average (3, 4, 4 meetings respectively).

Figure 5.9 Phone meetings and meeting management genre system in two phases



The decreasing use of meeting management genre system in phase II is in line with the finding presented in the previous chapter that scheduling phone meetings required less coordination as they became a routine. As members got used to the weekly meeting schedule, most regular meetings went as scheduled without explicit coordination, and the meeting management genre system was mainly enacted when they needed to change this regular rhythm, for example, to schedule an additional meeting or to reschedule.

Group authoring genre system

The group authoring genre system centers on the activities of authoring documents collaboratively. Whereas the three task specific e-mail genres and the meeting management genre system are for internal communication and coordination, the group authoring genre system concerns writing for external audiences. Like other organizations, LC produced various types of documents in multiple media – business letters, business plans, a white paper on their technology, legal documents, company web pages, presentation slides, etc. – to present their technology and organization to outsiders. Authoring these documents involved collaboration among members, from drafting to final editing.

Members mainly used emails to author documents collaboratively from a distance. Email seemed to be the appropriate medium for this purpose given its archiving and editing capabilities, with which members could easily track versions of drafts and add their comments alongside the circulated drafts. Members commented on the advantage of email in interviews:

> The advantage of email like me who saves everything is that once you've put it in email, you've got some piece of documentation that you can go back to...
>
> Email was useful when you wanted to exchange a lot of technical information. It was really useful for things like position papers when you're trying to hash strategy for building a product.

Moreover, authoring a document through emails resembled in many ways using emails for discussion, which members had been doing daily from the inception of the project. Usually a discussion was initiated by an email where a member presented his argument and invited others' feedback. The other members responded with their comments and these interactions went on until the topic was rested. Different from internal group discussions that could run indefinitely, however, documents had specific purposes associated with certain audience and deadlines. For example, a response email to a potential business partner had to be sent in a timely manner, presentation slides had to be prepared before the meeting, and a press release had to be published at the time of the product release. Authoring documents required more explicit coordination than general group discussions, and the group authoring genre system developed to coordinate this process through emails.

The first genre in the group authoring genre system is the *circulated draft*. Circulated draft emails typically consist of an introduction and a draft. The introduction part varied from just a subject line to a few lines in the body explaining the purpose and context of the draft. In many cases, a subject line sufficed and an explicit invitation to comment was also often omitted:

> Subject: proposed email to Adam Black
>
> Gentlemen -

I attach a proposed email message that I composed for Mr. Black using the input you gave last week. Please review this and make suggestions.

Fred

Hi Adam -

Remember that we sat together at the Awards reception at the conference last month? [draft continues]

The second genre in the group authoring genre system is the *reaction to draft*, by which members provided feedback to the circulated draft. Usually, these messages were nested beneath the draft itself, with all or part of the previous email embedded. Fred's draft email in the above example received three responses in less than one hour, one of which was Robert's reaction as shown below:

Subject: re: proposed email to Adam Black

Fred wrote:

> Gentlemen -
>
> I attach a proposed email message that I composed for Mr. Black using
> the input you gave last week. Please review this and make suggestions.
>
> --
> fred

1. fix the formatting of the document....your paragraphs are all on the same line.

2. in the section about [...] , we need to mention that the [...]

3. I think the most critical deficiency is that we are only on ms/intel platforms. I think we should list this as a deficiency and bluntly say that we would like to be on Linux and PowerPC and that we are interested in joint ventures to fund these ports.

--
robert

As seen in Robert's email, the comments provided by members ranged from the content to formatting and styling of the documents.

The *Revised draft* is the third genre in the group authoring genre system. After incorporating other members' suggestions, Fred sent out the revised version three days later, changing the subject line to "2nd try at email to ..." (see Table 5.4). As a draft usually underwent several rounds of revisions, it was a common practice to serialize subsequent versions to indicate the latest one. Revised draft is not much different from circulated draft in terms of form. A short description of the changes made to the previous version is followed by the invitation for comments.

The fourth genre, the *final version* of a document, appeared least frequently among the constituent genres of group authoring genre system. The final version of an outgoing email was simply sent to the recipient, with a copy sent to the rest of the team simultaneously. Sometimes, final versions were stored on the work server, with a notification email sent to members.

Table 5.4 Group authoring genre system enacted around authoring an email

Subject line	Genre	sender	Time sent
proposed email to Adam Black	Circulated draft	Fred	7/6/99 19:37:20
re: proposed email to Adam Black	Reaction to draft	Robert	7/6/99 19:59:06
re: proposed email to Adam Black	Reaction to draft	Keith	7/6/99 20:27:32
re: proposed email to Adam Black	Reaction to draft	Robert	7/6/99 20:33:33
2nd try at email to Black	Revised draft	Fred	7/9/99 00:06:25
re: 2nd try at email to Black	Reaction to draft	Robert	7/9/99 00:18:44
re: 2nd try at email to Black	Reaction to draft	Dan	7/9/99 00:30:47
re: 2nd try at email to Black	Reaction to draft	Keith	7/9/99 00:37:43

The interactions around enacting the group authoring genre system described so far appear similar to the common model of collaborative writing found in various business settings. Once members agreed to produce a document, a member either volunteered or was assigned to compose a draft based on various criteria, such as

expertise, availability, or existing connection to the recipient. This person usually coordinated the group authoring process, integrating comments from each round into subsequent revisions. However, LC's group authoring practice was more dynamic and flexible than the model. Sometimes a member other than the one who originally signed up for it would volunteer to compose the draft, or the person compiling subsequent versions would be different from the one who drafted it. In other words, the whole process was dynamically coordinated as a group rather than a linear process based on pre-defined roles.

The following example where LC members enacted the group authoring genre system to write a press release illustrates this. Preparing for public release of their product, LC decided to publish a press release. Initially, Fred was going to compose the first draft since all three full time members were occupied with various technical tasks to make the product ready. But as Fred's draft became delayed, Robert took up the task:

> Subject: press release
>
> Fred was going to write a press release, but he seems
> to be out today. Here is my try:
>
> LC ANNOUNCES THE AVAILABILITY OF THE....[draft continues]
> --
> robert

When Fred came back, he circulated a second draft based on members' reactions to Robert's first draft. From then on, Fred mainly coordinated the process summarizing necessary inputs and integrating them into the next version. But other members also occasionally circulated revised versions. Table 5.5 shows not only that all five members participated in authoring the press release but also that members took multiple roles (e.g., drafting, reacting, revising) in enacting the group authoring genre system.

Table 5.5 Group authoring genre system enacted around authoring a press release

Subject line	Genre	Sender	Date sent
press release	Circulated draft	*Robert*	*4/15/99*
re: press release	Reaction to draft	Dan	4/16/99
re: press release	Reaction to draft	Robert	4/16/99
re: press release	Reaction to draft	Dan	4/16/99
re: press release	Reaction to draft	Fred	4/19/99
2nd draft of press release	Revised draft	*Fred*	*4/23/99*
re: 2nd draft of press release	Reaction to draft	Robert	4/23/99
press release 2	Revised draft	Fred	4/23/99
re: press release 2	Reaction to draft	Robert	4/23/99
Try 3 for the press release	Revised draft	*Keith*	*4/23/99*
re: try 3 for the press release	Reaction to draft	Robert	4/23/99
Try 4 for the press release	Revised draft	*Keith*	*4/23/99*
re: try 3 for the press release	Reaction to draft	Dan	4/23/99
Try 5 for the press release	Revised draft	*Keith*	*4/23/99*
re: try 5 for the press release	Reaction to draft	Martin	4/23/99
re: try 5 for the press release	Reaction to draft	Robert	4/23/99
press release 6	Revised draft	*Fred*	*4/24/99*
re: press release 6	Reaction to draft	Robert	4/24/99
re: press release 6	Reaction to draft	Keith	4/24/99
re: press release 6	Reaction to draft	Robert	4/24/99
re: press release 6	Reaction to draft	Dan	4/24/99
re: press release 6	Reaction to draft	Robert	4/24/99
re: press release 6	Reaction to draft	Dan	4/24/99
re: press release 6	Reaction to draft	Fred	4/24/99
re: press release 6	Reaction to draft	Robert	4/24/99
press release 7?	Revised draft	*Dan*	4/24/99
re: press release 7?	Reaction to draft	Robert	4/24/99
re: press release 7?	Revised draft	*Fred*	4/25/99
press release	Revised draft	*Fred*	4/26/99
re: press release	Reaction to draft	Robert	4/26/99
re: press release	Reaction to draft	Dan	4/26/99
final press release	Final version	*Fred*	4/28/99

Authoring documents required a schedule. Whether it was associated with actual or perceived deadlines, authoring those documents could not be postponed too long for the sake of other work. When the document had a clear deadline, members planned it carefully. For example, during the week before LC's meeting with *Alpha*, a big player in the industry, LC held two internal phone meetings to prepare for the presentation (see Table 5.6). According to the meeting minutes below, members not only agreed to the

general division of work but also scheduled an additional meeting just one day before the presentation to go over the presentation slides:

> LC 28 January 1998 1:07 p.m. MST
>
> [....]
> Alpha wed morning 9 a.m.
> Call Tues and Wed, same time
> [...]
> Keith will create text presentation and send it around.
> Martin will turn this into Power Point.
> Fred will try our [system] with [Alpha's equipment].

As decided in the meeting, Keith circulated a draft two days later, and then the second draft before the next phone meeting. During the phone meeting, Keith gathered final suggestions for content and style and sent out the third draft right after the meeting. After the presentation, LC had another internal phone meeting to improve the slides for the future use. The next day Dan started another group authoring genre system by circulating an updated draft, and the final version from these interactions provided a backbone for later presentations.

Table 5.6 Group authoring genre system around authoring presentation slides

Subject line	Genre	Sender	Time
Phone meeting	Conference Call	All present	1/28/1998 13:07 MST
first draft of talk	Circulated draft	Keith	2/2/1998 23:11 EST
re: first draft of talk	Reaction to draft	Robert	2/2/1998 21:23 MST
re: first draft of talk	Reaction to draft	Martin	2/3/1998 0:31 PST
the talk	Revised draft	Keith	2/3/1998 13:29 EST
re: the talk	Reaction to draft	Robert	2/3/1998 11:34 MST
re: the talk	Reaction to draft	Robert	2/3/1998 11:54 MST
Phone meeting	Conference call	All present	2/3/1998 12:00 MST
new talk	Final draft	Keith	2/3/1998 15:01 EST

However, not every document was prepared well ahead of time. Sometimes, a document was needed at the last minute. See the example below where Martin asked for a flyer to distribute at the next day's exhibition:

> Subject: LC flyer
>
> I will have an exhibitor pass to [name of the exhibition]. It will allow me to walk around the exhibits. I thought that it would be great if we had a flyer that I could deposit at some, undetermined yet, strategic location(s).
> Did you talk about it? Did you want to do it?
> Do you have something I can use (1 sided 8.5"X11" page)?
> If we don't have anything could you give me a text of something by tonight and I'll make sure to duplicate it and distribute it?
>
> Martin

LC did not have anything ready for Martin to use, but Dan volunteered to prepare one right away and sent it to him.

Members frequently enacted more than one authoring genre system concurrently as they authored multiple documents at the same time, partly due to the irregularity of external communication: [32] For instance, members had to start composing a response to a time-sensitive matter, while they were still in the process of authoring a previous document. As a group authoring genre system could run for an extended period of time creating a long thread of emails, it was not rare that another case of the group authoring genre system interrupted the previous one. For example, the group authoring genre system enacted around the press release described earlier lasted 14 days consisted of 34 emails from all five members. During the two weeks, drafts of another two documents were also being created along with the press release.

In addition, group authoring activities of a similar type were often clustered into a certain period of time (e.g., conducting a series of sales presentations after a main product

[32] In using emails, LC members typically participated in concurrent threads. See Yates et al. (2003).

release). For example, during the one-month period from late February to mid March 1999, following LC's official rejection of a merger proposal, LC members enacted intense group authoring activities. During this period, LC drafted 11 documents, mostly business letters to attract major companies. It involved 75 emails, including almost one fifth (19.1%) of the total group authoring emails in phase II. Members were engaged in at least two concurrent cases of group authoring genre systems at any given moment, with an average elapsed time of 1 day 8 hours and 19 minutes.

Comparison between two phases

There is a substantial increase in the use of the group authoring genre system between phase I and phase II. In phase I, LC members enacted only 16 cases of the group authoring genre system, but they enacted 83 cases in phase II. Similarly, the number of messages in group authoring genre system increased. These messages accounted for only 1.4% (48 messages) of the total messages in phase I, but 13.3% (393 messages) in phase II. It reflects the increased frequency of external communication in phase II in expanding the company's network to market the product. As a result, the types of documents and external communication partners became more diversified in phase II. However, the use pattern of constituent genres is not much different between the two phases. LC members were very active in responding to a circulated draft. Combining reactions to drafts and revised drafts, they sent around twice (0.4% vs. 0.8%) and slightly over four times (2.6% vs. 9.9%) as many reaction emails as circulated draft emails in phase I and phase II respectively. As mentioned earlier, final drafts appeared very rarely (0.2% and 0.8%).

Only about half (7 out of 16) of the cases in phase I and slightly over one third (25 out of 83) of the cases in phase II had final versions circulated by email.

Figure 5.10 Use of group authoring system

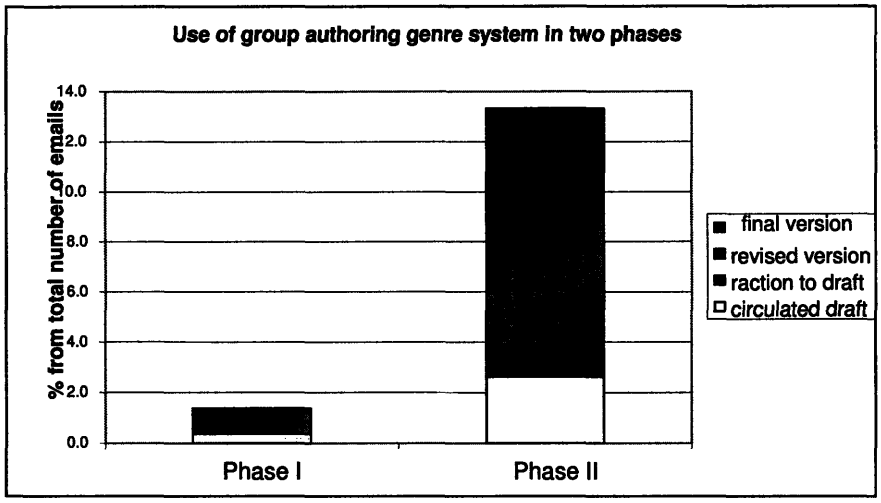
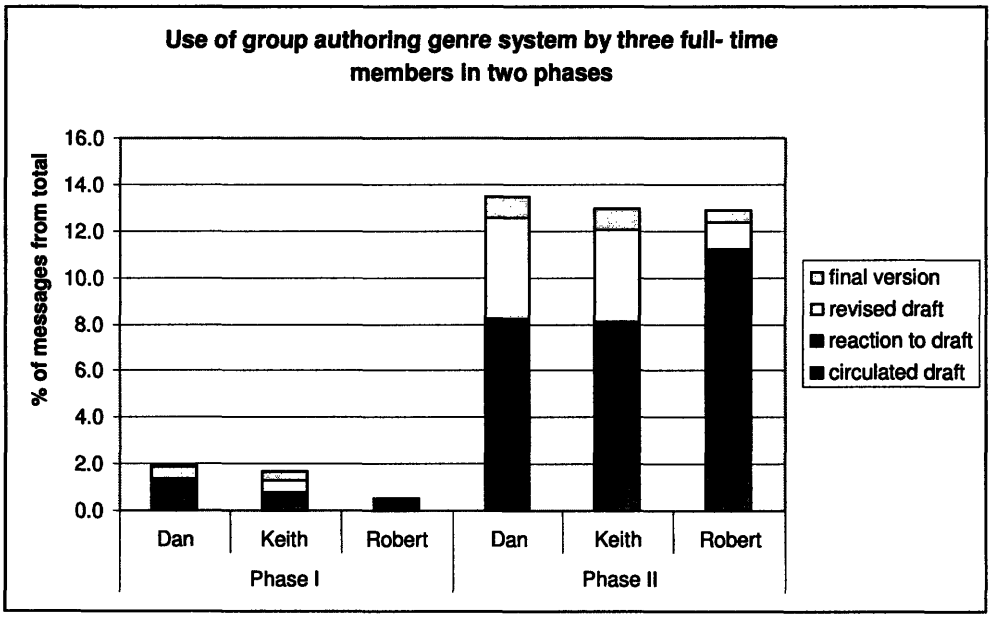


Figure 5.11 Use of group authoring genre system by full time members



The role of the three full-time members was central in this genre system. In phase I, Robert's participation was relatively minor compared to that of the other two, because the majority (56.3%) of the group authoring cases were around composing emails to a few external experts on technical issues that concerned Dan and Keith's work, which was handled by the two members without Robert's direct involvement. In contrast, as external communication in phase II was less about specific technical issues and more about general business and marketing, the areas in which none of the three full-time members had superior expertise and thus all were equally responsible, their participation in the group authoring genre system became more evenly distributed.

Figure 5.12 shows the trend in the use of group authoring emails over time in two phases. The group authoring genre system, unlike other genres in LC, shows an intermittent use pattern, for reasons already explained. Even in phase II, the group authoring genre system was enacted sporadically, showing periods of intensive use interspersed with intervals of non-use.

Figure 5.12 Monthly use of group authoring genre system in two phases

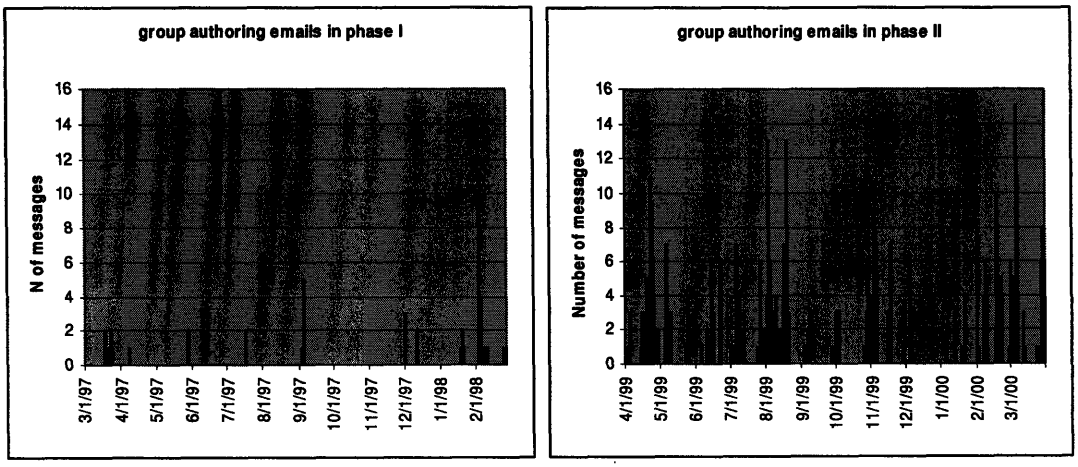


Table 5.7 Summary of the group authoring genre system in two phases

	Phase I		Phase II	
Number of GA genre systems enacted	16 cases		83 cases	
Average number of messages/thread	3 message		4.7 messages	
Median number of messages/thread	2 messages		3 messages	
Maximum number of messages/thread	8 messages		34 messages	
Minimum number of messages/thread	1 message		1 message	
Average elapsed time for a thread	7d 23h 21m 33s		1d 10h 3m 36s	
Median elapsed time for a thread	7h 19m 14s		2h 53m 53s	
Maximum elapsed time for a thread	50d 53m 48s		16d 14h 4m 53s	
Minimum elapsed time for a thread	15m 31s		5m 23s	
Average number of concurrent GA threads	1.9		1	
Median number of concurrent GA threads	1.5		1	
Maximum number of concurrent GA threads	5		8	
% of more than two concurrent threads	50%		43.4%	
	N	%	N	%
Number of messages in GA genre system†	48	1.4%	393	13.3%
Circulated draft†	13	0.4%	78	2.6%
Reaction to draft†	28	0.8%	290	9.8%
Final version†	7	0.2%	25	0.8%
Cases with final version	7	43.8%	25	30.1%
Orphaned drafts	3	6.3%	14	16.9%

† percent from the total number of messages in each phase.

The pace of authoring documents accelerated in phase II. In spite of the higher number of messages per thread, which increased on average from 3 in phase I to 4.7 in phase II, the elapsed time for a thread decreased (8 day to 1.5 day on average and 50 days to 16.5 days in maximum) (see Table 5.7). This increase in pace also explains why the average number of concurrent threads actually decreased in phase II, although members enacted eight group authoring genre systems concurrently at one point. As the project entered post-development phase, authoring documents related to marketing and sales activities became more central and members concentrated on these tasks with more

intensity. The accumulated knowledge in the forms, styles and contents of different types of documents through repeated enactments also contributed to the accelerated pace.

Conclusion

Focusing on the genres and genre systems enacted by LC members to structure their email communication, I have showed how virtual team members can use their communication to accomplish the temporal coordination of their distributed activities. Within LC, the three genres (status report genre, bug/error notification genre, update notification genre) and two genre systems (phone meeting management genre system and group authoring genre system) were identified as key communication structures that both reflected and shaped members' temporal and work practices. Through enacting task-specific genres, members shared important temporal information that allowed them to schedule individual activities, synchronize efforts, and allocate tasks. The phone meeting management genre system not only demonstrates the dynamic and negotiated nature of temporal structuring in LC, but also shows that a strong organizational temporal structure can emerge through such processes. The group authoring genre system helped coordinate the process of collaborative writing and produce documents in timely manner. The habitual enactment of these genres and genre systems through emails gave coherence and continuity to their dispersed activities.

The findings further confirm the conclusion from the previous chapter that structuring communication is an important means of temporal coordination in distributed teams. In particular, I have tried to show that the notions of genre and genre system are

particularly useful to understand and analyze how such structuring actually occurs over time. These concepts shift the research focus from how a given set of temporal coordination mechanisms affect team performance to how they emerge, stabilize, and change over time. Most of prior research on temporal coordination in virtual teams has stressed the importance of formal temporal structures for better coordination and performance within virtual teams. However, in real settings, imposing temporal structures at the outset is often neither possible nor practical. In many cases, such formal temporal structures are too abstract to guide everyday activities or too rigid to accommodate different and dynamic local situations. The examination of LC members' ongoing email communication suggests that their recurrent use of specific genres and genre systems affords them such dynamic and flexible coordination. They structure communication around shared temporal expectations and norms, but are flexible enough to allow members opportunities for negotiation and adjustment. Furthermore, as organizing templates emergent from internal communication, genres and genre systems evolve and change as members modify them or invent new ones in response to changing situations, as I have shown by tracking the change in form and use of each genre and genre system in two phases.

A focus on genres and genre systems is also useful in understanding how virtual teams create shared temporal structures and rhythms that help coordinate dispersed activities. Reddy and Dourish (2002) note that knowledge about organizational temporal rhythms is an important resource for workers to accomplish their work and guide their future activities. In contrast to the co-located medical workers in their study, whose main coordination challenge was to learn and adapt to institutionalized temporal rhythms,

virtual team members have to actively produce such knowledge and make it "visible" to the rest of the team through communication. Genres and genre systems facilitate this communication, helping members make critical information available "at the right place at the right time," where the "rightness" is embedded in different genre norms established implicitly or explicitly over time. In the LC, for example, the update notification messages were expected to be sent almost simultaneously with the actual update, whereas the status report was less tightly linked to a particular moment in time. The content and timing of these genres provided critical information to members, allowing them to shape and adapt their own work rhythms. Furthermore, enacting genres and genre systems also contributed to creating more routine organizational rhythms, as seen in the example of the meeting management genre system in LC. Genres and genre systems, therefore, help temporal coordination by directly or indirectly generating collective rhythms, not in the sense that they are enacted on a fixed schedule, but in the sense that they shape expectations regarding a team's collective work rhythms to which individual members can orient their own. Through enacting genres and genre systems, LC members constantly provided each other opportunities to check discrepancies and synchronize their distributed efforts.

The implications are not limited to the topic of temporal coordination. Some researchers have observed that temporal issues (eg., deadlines, schedules) often create second agendas that contributed to the general improvement of group processes by redirecting members' attention from the principal activities of the team to more process oriented tasks (Gersick, 1988;1989; Okhuysen and Eisenhardt, 2002). Genres and genre systems function as vehicle for members to bring second agendas into ongoing discussion

and negotiation. While temporally coordinating their work through enacting genres, members inevitably attend to other issues that go beyond temporal issues. For example, the bug/error notification genre not only notifies members when a bug or an error has been found and how urgently a solution is needed, but typically also suggests who should take care of the job and how best to proceed. By adding second agendas as a critical part of daily communication, genres and genre systems can contribute to enhancing members' awareness of issues beyond the primary tasks, thus improving overall group processes.

The genres and genre systems in LC emerged and developed over time as members attempted to structure their communication in recurrent situations originating from their software development work in a distributed context. I expect the critical role of genres and genre systems in temporal structuring and coordination to be evident in other types of distributed or virtual organizations. However, the actual genres/genre systems in use may differ across teams and organizations, depending on differences in goals, tasks, technologies in use, and members' local situations.

CHAPTER 6
Conclusion

The rapid expansion of distributed work raises an important question: how do people working across different locations and time zones through technological mediation coordinate and align their activities? In this dissertation, I followed the four-and-a-half-year journey of a small software team operating almost entirely virtually to examine what people actually do on a daily basis to make the distributed, technologically mediated collaboration possible. The findings from LC neither celebrate the transition to this new mode of work for the increased flexibility of "working any time any place" nor denounce it for its negative effects on team effectiveness. Rather, they bring to the fore what is hidden behind the banner of virtual work, the ongoing practices of team members to actualize the potential of the technology and meet the unforeseen challenges. I have demonstrated how much planning, negotiating, adapting, managing, rearranging, and juggling was required in order for members to work flexibly yet stay synchronized, to make communication run smoothly yet dynamically integrate different forms of mediated communication for appropriate coordination, to establish shared rules and norms for stability yet modify them in the evolving contexts.

In chapter three, I examined the temporal practices in LC— how LC members experienced, used, and managed their time—based on their communication data and interviews. The results show that LC members had a high level of operational flexibility, with freedom to choose when to work or how many hours to work on a particular day. Although their actual work hours are hardly accessible, statements from members as well as their communication data suggest their work might have been as intense as that of

other software teams. I argued that LC members' estimate of 50-hour-week is thus the symbolic form of their temporal standards and norms, which represents the level of productivity and commitment that members deemed appropriate for highly skilled technical professionals—that is, their identity translated into temporal terms.

For the individual members who worked from home, flexible work first required them to find their own temporal rhythms and structures both to refrain from incessant work as well as to enable consistent work. It involved blurring boundaries between various activities that used to be separated into different realms of life in a collocated situation (e.g., home and work) as well as constructing new ones. As each member's sense of boundary was different, LC members had to achieve boundary management as group. In the absence of organizational boundary control, this achievement involved negotiating individual flexibilities through which members' different temporal expectations, interests, and patterns were made visible. Once they learned others' temporal structures over time, much of the boundary management was done implicitly through mutual adjustment. Creating and establishing an organizational routine, such as weekly phone meetings, facilitated coordinating flexible schedules of individual members by providing a temporal anchor in LC.

Flexibility in LC was socially accomplished through mutual adjustments, negotiations, or sometimes conflicts among members. It took time and effort for LC members to learn how to put flexibility into useful practices, both in structuring their work day and collaborating with each other. Flexibility derives not from technology or from work itself but from the innovations in social relations and practices of people who actually do the work.

Chapter four delves into the social practices of LC members that enabled flexibility. Combining the insights from prior research that temporal norms and standards in organizations are closely related to the way work is coordinated, that coordinating interdependent work presupposes awareness of the tasks and the team, and that maintaining such awareness depends on communication, I explored the connections between communication, coordination, and flexibility. First, I demonstrated that LC members also struggled with the challenges of maintaining awareness of the task and the team in distributed situation, facing various mutual knowledge problems. Learning from doing, LC members gradually developed useful coordination practices over time, in which structuring and adapting different forms of mediated communication played a critical part.

LC's coordination practices are characterized by interweaving the personal and group modes of coordination according to the tasks and situations. For tasks that needed the expertise of a particular individual and close co-work with that person, they used the personal mode of coordination through dyadic communication. For tasks that required general expertise or that affected everyone in the team, they used the group mode of coordination through group emails or phone meetings. As the task situation changed, the switches between two modes were frequent. They not only coupled types of communication with the needed coordination, but also maintained the balance across different coordination types to achieve both efficiency and awareness. Members also increased communication and coordination of all types in periods of high activity. The findings from LC demonstrate that coordination emerges from the interactions among the team members. It was particularly so in software development, where tasks are highly

interdependent yet somewhat unpredictable and thus hard to plan ahead. When work itself was emergent, so was its communication and coordination. The adaptive evolution of communication and coordination practices found in LC highlights that communication and coordination are not static team processes but dynamic capabilities members learn and develop over time.

I also have shown that this dynamic coordination derives partly from how the team was constituted in terms of expertise. Members' specialties allowed them to work autonomously on modularized tasks, adjusting their work schedules more flexibly to their local situations. At the same time, the broad base of overlapping expertise among members made the coordination a function of the group in general, not the role of particular individuals. The group mode of coordination was dominant and effective in LC, bridging possible gaps of mutual knowledge that might have been caused by the personal mode of coordination, because members were capable of understanding the task situation in other locations without paying too much time and learning cost. The group mode of coordination, then, facilitated dynamic distribution of tasks beyond the preplanned division of work, allowing members to broaden their skills into areas beyond their pre-existing specialties. As members became more versatile, the group mode of coordination became more effective, which further ensured flexibility of when members chose to work on what. However, the same group mode of coordination did not incur this positive cycle in "non-technical" tasks where members lacked basic expertise and experience. LC's group coordination practices highlight that the way work is coordinated affects the temporal norms and practices in organizations, yet the influence is not simple, moderated by other factors, such as expertise.

Chapter five goes further to explore the interconnections among communicating, coordinating, and temporal structuring in LC by focusing on the genres and genre systems enacted by LC members to structure their email communication. Within LC, the three genres (status report genre, bug/error notification genre, update notification genre) and two genre systems (phone meeting management genre system and group authoring genre system) were identified as key communication structures that both reflected and shaped members' temporal and work practices. Through enacting task-specific genres, members shared important temporal information that allowed them to schedule individual activities, synchronize efforts, and allocate tasks. The phone meeting management genre system not only demonstrates the dynamic and negotiated nature of temporal structuring in LC, but also shows that a strong organizational temporal rhythm can emerge through such processes. The group authoring genre system helped coordinate the process of collaborative writing and produce documents in a timely manner. The habitual enactment of these genres and genre systems through emails gave coherence and continuity to their dispersed activities.

Most prior research on virtual teams emphasizes the importance of structuring communication and work for temporal coordination, but little addresses how such structuring actually occurs over time. The notion of genre and genre system provides useful analytical tools to understand and analyze this structuring process through communication. Genres and genre systems are "vehicle" for members to transmit critical information "at the right place at the right time," where the "rightness" is embedded in various genre norms. Each enactment is also a "message," the content and timing of which guides members to shape and adapt their own rhythms. Furthermore, enacting

genres and genre systems also contributes to creating more routine organizational rhythms, as seen in the example of the meeting management genre system in LC. Genres and genre systems, therefore, directly or indirectly generate collective rhythms, not in the sense that they are enacted on a fixed schedule, but in the sense that they shape expectations regarding a team's collective work rhythms to which individual members can orient their own. Through enacting genres and genre systems, LC members constantly provided each other opportunities to check discrepancies and synchronize their distributed efforts.

I also attempted to extend understanding of temporal coordination in virtual teams by shifting focus from how a given set of temporal coordination "mechanisms" (or the lack thereof) affect team performance to how they emerge, stabilize, and change over time. As I demonstrated in chapter four, in a real setting where work is emergent and evolving, formal temporal structures are too abstract to guide everyday activities or too rigid to accommodate different local situations. The examination of LC members' ongoing email communication suggests that their recurrent use of specific genres and genre systems affords such dynamic and flexible coordination. They structure communication around shared temporal expectations and norms, but are flexible enough to allow members opportunities for negotiation and adjustment. Emergent from members' interactions, genres and genre systems evolve and change as members modify them or invent new ones in response to changing situations. It enables us to observe both stability and change in communication practices, which also signals related reconfiguration of coordination and temporal practices.

Communication is fundamental to any form of organizing, but it is preeminent in virtual organizing. Without it, the boundary spanning activities of distributed team cannot be possible to begin with, and distributed collaboration cannot go on. Given the centrality of communication in virtual organizing, a meaningful study of one requires an account of the other. The rich communication data from LC provided the opportunity to deepen our understanding of the communication processes in distributed teams. Moreover, given the recursive relationships between time, communication, and work I discussed in the introduction, it opened a window to explore other critical aspects of virtual organizing—in particular, temporal and work practices. In short, by examining their communication, I could access the "communicative structuring" of time and work in LC.

By focusing on communicative structuring, I showed that time, communication, and coordination are all implicated in everyday practices members enact to synchronize their efforts, stay connected, and work together in distributed situation. For example, LC members were communicating, coordinating, and temporally structuring their work simultaneously while enacting genres and genre systems in email. Of course, this is not unique in virtual or distributed situation. Although it may sound paradoxical, "disembedding" (Giddens, 1991) social relations from the common contexts of interactions such as time, place, and sequence of conversations makes their enacted nature and mutual implication even more palpable to dispersed individuals. Examples are abundant in LC. Instead of simply leaving an office to signal the end of the day, LC members had to "check out" by sending an email saying "I'm going to bed now." As they could not glance over the other member's computer screen, members had to translate it "as it happened" so that it could be reproduced in other locations. "3 p.m." ceased to be

an apparent temporal reference to the members in three different time zones, so time had to be standardized. In short, compared to the collocated situation where the common reference points make communicating, coordinating, and temporal structuring appear more effortless, less problematic, and thus "natural," distributed work demands members to constantly engage in contextualizing (embedding one's context), translating (deciphering others' context and adapting one's own), and standardizing (creating shared context) to make them happen.

Time, communication, and coordination are not only simultaneously implicated in every practice, but also dynamically reconfigured in evolving work, social relations, and local contexts. In chapter four, for example, I demonstrated that the adaptive configuration of communication and coordination practices facilitated flexibility in LC. In doing so, I showed that flexibility, communication, and coordination are not static team processes but dynamic capabilities that members have to learn over time. It is this "situated learning" (Lave and Wenger, 1991), which involves negotiations, mutual adaptations, and sometimes conflicts, through which members become a competent participants on the team. Seen in this light, a "virtual team" is not a unified, static entity that triggers communication and collaboration but is also the outgrowth of them.[33] Technology may connect people in remote locations, but that is only a start. In fact, maintaining interconnectedness through technology takes constant work from dispersed individuals, as described in chapter two. Technologies may assemble people into a virtual team, but what constitutes them as a team are a wide range of meticulous strategies, techniques, and social practices members create, develop, and share over time.

[33] This does not mean that traditional teams are homogeneous and unified units, but virtual teams surely provide a metaphor that promotes our thinking of teams with renewed emphasis on highly dynamic processes, permeable boundaries, and reconfigurable structures (DeSanctis and Monge, 1999).

This is one of the reasons why it is so important to study virtual organizing in its real setting over an extended period of time. Most prior research on virtual teams based on experimental settings or short-term observation misses the important dynamic that unfolds over time. The mixed results so far from research on the effects of geographic dispersion on team performance can be partly attributed to this limitation. Team performance depends on the team processes such as communication and coordination, as properly recognized by most research, but such team processes are not static parameters of teams that can be taken out of the contexts and directly compared. Rather, they emerge from the shared experience members gradually accumulate as a team. The adaptive reconfigurations of these processes in LC are from "learning-in-virtual-organizing." Even in a small team like LC, it took substantial time to learn how to effectively collaborate as a team. As teams are increasingly spanning more boundaries across time, space, organizations, and cultures, it becomes more important to understand how people learn to bridge heterogeneous, diverse, and locally situated practices.[34]

This dissertation contributes to the existing body of virtual team research by providing a longitudinal case study that focuses on actual practices of a distributed team working together to meet deadlines and develop a new product. By examining a real team carrying out a particular type of work (software development) over an extended period of time, this dissertation also has implications for various streams of research, including studies on time, communication, and coordination in teams and organizations.

[34] While many researchers examined situated learning in a variety of settings, most have focused on the work practices of individuals (Lave and Wenger, 1991; Orr, 1996; Suchman, 1987) or that of groups proximate in time and space (Cook and Yanow, 1996; Pentland, 1995; Weick and Roberts, 1993). Considering the emphasis on embodied and embedded practices in situated learning theory, research should extend to how virtual teams situated in diverse contexts and mediated through technology achieve such learning (for a rare example, see Orlikowski and Yates, 2002).

The importance of situated practices and local contexts in virtual organizing, which makes the findings reported here meaningful, also requires some caution. LC was a small team composed of members who had high commonality (in age, gender, nationality, educational and professional background etc.), and some of them had social ties before they came into the team. It gave LC some advantage over teams composed of people with few prior social relationships,[35] particularly in terms of mutual understanding and trust that are known to be difficult to develop and maintain in virtual settings.[36] Also virtual teams of a bigger size spanning more boundaries (e.g., cross-cultural teams), and with more media to choose from may reveal more interesting dynamics. In addition, as teams are typically engaged in boundary spanning activities with other teams, departments, organizations, and communities, they may have to adapt their existing practices for such interactions. For example, email can be an effective means of communication among team members but can be interpreted differently by an outside communication partner (e.g., lack of trust, low priority). In this dissertation, I focused mainly on the internal communication and coordination in the team of software engineers, for which LC's communication data is most informative. Although it is beyond the scope of this dissertation, there are clues of changes in LC after Ray, new CEO of the company, joined the team. Ray provided an interesting comparison in the interview:

> They [Dan, Keith and Robert] showed me a list of 600 people [who downloaded their software product]. I said 'how many have you spoken to?' [They answered] 'Oh, we email to a couple.' Their primary way of communication is email. Picking up the phone, from a technical person is not what they would rather do. It's not their preference at all. [...] I've assigned each one of them as a technical

[35] See Moon and Sproull (2002) on the Linux community, for example.

[36] A research on global virtual teams reported virtual teams form "swift trust" but such trust is fragile and temporary (Jarvenpaa and Leidner, 1999).

> support liaison to the clients. They had to pick up the phone and speak to these people, even when it comes to the point of, 'well, we really can't help you.'

Coming from the marketing and sales background—"he is not one of us," aptly pointed out one of the full-time members—Ray introduced some changes in internal team processes as well as external communication. As the team was reconfigured, both Ray and the three software engineers had to learn how to communicate and coordinate effectively as a team. "There has been a two-way learning curve," commented one of the LC members, again highlighting the adaptive reconfigurations of team processes in virtual teams. As virtual teams are increasingly composed of members with diverse backgrounds, often reconstituted during the team's existence, and engage with various external communication and coordination, studying how virtual teams create "pidgins and creoles" in "trading zones" (Galison, 1997) of heterogeneous local practices will gain much importance.

In sum, while LC is a particular setting, I believe that the findings presented here offer insights for future research in other empirical contexts. In particular, I invite more longitudinal studies on virtual teams in different contexts to examine and elaborate the findings from this dissertation.

References

Adam, B. 1990. *Time and Social Theory*. Philadelphia: Temple University Press.

Adam, B. 1995. *Timewatch: The Social Analysis of Time*. Cambridge, MA: Polity Press.

Ancona, D. G., and Caldwell, D. F. 1992. "Demography and Design - Predictors of New Product Team Performance." *Organization Science* 3, 3, 321-341.

Ancona, D. G., and Chong, C. L. 1996. "Entrainment: Pace, Cycle, and Rhythm in Organizational Behavior." In B. Staw and L. L. Cumming (eds.) *Research in Organizational Behavior*. Greenwich: JAI Press, 151-184.

Ancona, D. G., Goodman, P. S., Lawrence, B. S., and Tushman, M. L. 2001. "Time: A New Research Lens." *Academy of Management Review* 26, 4, 645-663.

Argote, L. 1982. "Input Undertainty and Organizational Coordination in Hospital Emergency." *Administrative Science Quarterly* 27, 3, 420-434.

Armstrong, D., and Cole, P. 2002. "Managing Distances and Differences in Geographically Distributed Work Groups." In P. Hinds and S. Kiesler (eds.) *Distributed Work*. Cambridge, MA: MIT Press, 167-186.

Arthur, M. B. 1994. "The Boundaryless Career - a New Perspective for Organizational Inquiry." *Journal of Organizational Behavior* 15, 4, 295-306.

Bailyn, L. 1985. "Autonomy in the Industrial R-and-D Lab." *Human Resource Management* 24, 2, 129-146.

Ballard, D. I., and Seibold, D. R. 2003. "Communicating and Organizing in Time: A Meso-Level Model of Organizational Temporality." *Management Communication Quarterly* 16, 3, 380-415.

Barker, J. R. 1993. "Tightening the Iron Cage - Concertive Control in Self-Managing Teams." *Administrative Science Quarterly* 38, 3, 408-437.

Barley, S. R. 1988. "On Technology, Time and Social Order: Technically Induced Change in the Temporal Organization of Radiological Work." In F. A. Dubinskas (ed.) *Making Time: Ethnographies of High Technology Organizations*. Philadelphia: Temple University Press, 123-169.

Barley, S. R., and Kunda, G. 1992. "Design and Devotion - Surges of Rational and Normative Ideologies of Control in Managerial Discourse." *Administrative Science Quarterly* 37, 3, 363-399.

Bash, H. H. 2000. "A Sense of Time - Temporality and Historicity in Sociological Inquiry." *Time & Society* 9, 2-3, 187-204.

Bassman, M., McGary, F., and Pajerski, R. 1994. "Software Measurement Guidebook." NASA.

Bazerman, C. 1994. "Systems of Genres and the Enactment of Social Intentions." In A. Freedman and P. Medway (eds.) *Genre and the New Rhetoric*. London: Taylor & Francis, 79-101.

Bennett, M., and Weill, P. 1997. "Exploring the Use of Electronic Messaging Infrastructure: The Case of a Telecommunications Firm." *Journal of Strategic Information Systems* 6, 7-34.

Bluedorn, A. C., and Denhardt, R. B. 1988. "Time and Organizations." *Journal of Management Information Systems* 14, 299-320.

Blyton, P., Hassard, J., Hill, S., and Starkey, K., eds. 1992. *Time, Work, and Organization*. London: Routledge.

Brockhoff, K. 1998. *Internationalization of Research and Development*. New York: Springer.

Cairncross, F. 1997. *The Death of Distance: How the Communications Revolution Will Change Our Lives*. Boston, MA: Harvard Business School Press.

Carmel, A. 1999. *Global Software Teams: Collaborating across Borders and Time Zones*. Upper Saddle River, NJ: Prentice Hall.

Cavendish, R. 1982. *Women on the Line*. London: Routledge & Kegan Paul.

Clark, P. A. 1978. "Temporal Innovation and Time Structuring in Large Organizations." In J. T. Fraser, N. Lawrence and D. Park (eds.) *The Study of Time*. New York: Springer-Verlag.

Clark, P. A. 1985. "A Review of the Theories of Time and Structure for Organizational Sociology." *Research in the Sociology of Organizations* 4, 35-79.

Clark, P. A. 1990. "Chronological Codes and Organizational Analysis." In J. Hassard and D. Pym (eds.) *The Theory and Philosophy of Organizations: Critical Issues and New Perspectives*. London: Routledge, 137-163.

Clarke, A., and Fujimura, J. H. 1992. *The Right Tools for the Job : At Work in Twentieth-Century Life Sciences*. Princeton, N.J.: Princeton University Press.

Cook, S. D. N., and Yanow, D. 1996. "Culture and Organizational Learning." In L. S. Sproull (ed.) *Organizational Learning*. Thousand Oaks, CA: Sage Publications.

Coser, L., and Coser, R. 1963. "Time Perspective and Social Structure." In A. W. Gouldner and H. P. Gouldner (eds.) *Modern Sociology*. New York: Harcourt, Brace, and World.

Cramton, C. D. 2001. "The Mutual Knowledge Problem and Its Consequences for Dispersed Collaboration." *Organization Science* 12, 3, 346-371.

Cramton, C. D. 2002. "Finding Common Ground in Dispersed Collaboration." *Organizational Dynamics* 30, 4, 356-367.

Curtis, B., Krasner, H., and Iscoe, N. 1988. "A Field Study of the Software Design Process for Large Systems." *Communications of the ACM* 31, 11, 1268-1287.

DeSanctis, G., and Monge, P. 1999. "Introduction: Communication Processes for Virtual Organizations." *Organization Science (Special Issue)* 10, 693-703.

Ditton, J. 1979. "Baking Time." *Sociological Review* 27, 157-167.

Dourish, P., and Bellotti, V. 1992. "Awareness and Coordination in Shared Workspace." In *Proceedings of CSCW'92*, 107-114.

Dubinskas, F. A. 1988a. "Cultural Construction: The Many Faces of Time." In A. Dubinskas Frank (ed.) *Making Time : Ethnographies of High-Technology Organizations*. Philadelphia: Temple University Press, 3-38.

Dubinskas, F. A. 1988b. "Janus Organizations: Scientists and Managers in Genetic Engineering Firms." In F. Durbinskas (ed.) *Making Time: Ethnographies of High-Technology Organizations*. Philadelphia: Temple University Press, 170-232.

Durkheim, E. 1976. *The Elementary Forms of the Religious Life, (2nd Ed.)*. London: George Allen & Unwin.

Edwards, R. C. 1979. *Contested Terrain*. New York: Harper & Row.

Evans, J. A., and Barley, S. R. 2004. "Beach Time, Bridge and Billable Time Hours: The Temporal Structure of Technical Contracting." *Administrative Science Quarterly* 49, 1, 1-38.

Evans, J. S. 1991. "Strategic Flexibility for High Technology Maneuvers: A Conceptual Framework." *Journal of Management Studies* 28, 1, 69-89.

Galegher, J., and Kraut, R. E. 1994. "Computer-Mediated Communication for Intellectual Teamwork: An Experiment in Group Writing." *Information Systems Research* 5, 2, 110-138.

Galison, P. L. 1997. *Image and Logic : A Material Culture of Microphysics*. Chicago, IL: University of Chicago Press.

Gambetta, D., ed. 1988. *Trust: Making and Breaking Cooperative Relations*. Oxford: Basil Blackwell.

Gareis, K. C., and Barnett, R. C. 2002. "Under What Conditions Do Long Work Hours Affect Psychological Distress? A Study of Full-Time and Reduced-Hours Female Doctors." *Work and Occupations* 29, 4, 483-497.

Gersick, C. J. G. 1988. "Time and Transition in Work Teams - toward a New Model of Group Development." *Academy of Management Journal* 31, 1, 9-41.

Gersick, C. J. G. 1989. "Marking Time - Predictable Transitions in Task Groups." *Academy of Management Journal* 32, 2, 274-309.

Gersick, C. J. G. 1994. "Pacing Strategic Change: The Case of a New Venture." *Academy of Management Review* 37, 1, 9-45.

Ghosh, T., Yates, J., and Orlikowski, W. J. 2004. "Using Communication Norms for Coordination: Evidence from a Distributed Team." in *Proceedings of the International Conference on Information Systems*, Washington, DC.

Giddens, A. 1984. *The Constitution of Society*. Berkeley, CA: University of California Press.

Giddens, A. 1991. *The Consequences of Modernity*. Cambridge, UK: Polity Press.

Golden, W., and Powell, P. 2000. "Towards a Definition of Flexibility: In Search of the Holy Grail?" *Omega: The International Journal of Management Science* 28, 373-384.

Gupta, P. P., Dirsmith, M. W., and Fogarty, T. J. 1994. "Coordination and Control in a Government Agency: Contingency and Institutional Theory Perspectives on Gao Audits." *Administrative Science Quarterly* 39, 2, 264-284.

Gutwin, C., Roseman, M., and Greenberg, S. 1996. "A Usability Study of Awareness Widgets in a Shared Workspace Groupware System." In *Proceedings of CSCW '96*, 258-267.

Hall, E. T. 1983. *The Dance of Life: The Other Dimension of Time*. Garden City, NY: Anchor Press/Doubleday.

Hall, E. T. 1990. *Understanding Cultural Differences*. Yarmouth: Intercultural Press.

Hassard, J. 1990. "Introduction: The Sociological Study of Time." In J. Hassard (ed.) *The Sociology of Time*. New York: St Martins Press, 1-18.

Hassard, J. 1996. "Images of Time in Work and Organization." In S. R. Clegg, C. Hardy and W. R. Nord (eds.) *Handbook of Organization Studies*. Thousand Oaks: Sage, 581-598.

Hecker, D. 1998. "How Hours of Work Affect Occupational Earnings." *Monthly Labor Review* 121, 10, 8-18.

Herbsleb, J., and Grinter, R. 1999. "Splitting the Organization and Integrating the Code: Conway's Law Revisited." In *Proceedings of the 21st International Conference on Software Engineering*, p. 85-95.

Hightower, R., and Sayeed, L. 1995. "The Impact of Computer-Mediated Communication Systems on Biased Group Discussion." *Computers in Human Behavior* 11, 1, 33-44.

Hinds, P. J., and Kiesler, S., eds. 2002. *Distributed Work.* Cambridge, MA: MIT Press.

Hutchins, E. 1990. "The Technology of Team Navigation." In J. Galegher, R. E. Kraut and C. Egido (eds.) *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work.* Hillsdale, NJ: Erlbaum, 191-220.

Im, H., Yates, J., and Orlikowski, W. 2005. "Temporal Coordination through Communication: Using Genres in a Virtual Start-up Organization." *Information Technology & People* 18, 2, 89-119.

Jacques, E. 1982. *The Form of Time.* London: Heinemann.

Jarvenpaa, S. L., and Leidner, D. E. 1999. "Communication and Trust in Global Virtual Teams." *Organization Science* 10, 6, 791-815.

Kanter, R. M. 1977. *Men and Women of the Corporation.* New York: Basic Books.

Kanter, R. M. 1989. *When Giants Learn to Dance : Mastering the Challenge of Strategy, Management, and Careers in the 1990s.* New York: Simon and Schuster.

Kidder, T. 1981. *The Soul of a New Machine.* Boston: Little Brown.

King, J. L., and Frost, R. L. 2002. "Managing Distance over Time: The Evolution of Technologies of Dis/Ambiguation." In P. Hinds and S. Kiesler (eds.) *Distributed Work.* Cambridge, MA: MIT Press, 3-26.

Kraut, R., Lewis, S., and Swezey, L. 1982. "Listener Responsiveness and the Coordination of Conversation." *Journal of Personality and Social Psychology* 43, 718-731.

Kraut, R. E., Egido, C., and Galegher, J. 1990. "Patterns of Contact and Communications in Scientific Research Collaboration." In J. Galegher, R. E. Kraut and C. Egido (eds.) *Intellectual Teamwork: Social and Technological Foundations of Cooperative Work.* Hillsdale, NJ: Erlbaum.

Kraut, R. E., Fussell, S. R., Brennan, S. E., and Siegel, J. 2002. "Understanding Effects of Proximity on Collaboration : Implications for Technologies to Support Remote

Collaborative Work." In P. Hinds and S. Kiesler (eds.) *Distributed Work.* Cambridge, MA: MIT Press, 137-162.

Kunda, G. 1992. *Engineering Culture : Control and Commitment in a High-Tech Corporation.* Philadelphia: Temple University Press.

Lave, J., and Wenger, E. 1991. *Situated Learning : Legitimate Peripheral Participation.* Cambridge, UK/New York: Cambridge University Press.

Liang, D., Moreland, R., and Argote, L. 1995. "Group Versus Individual Training and Group Performance: The Mediating Factor of Transactive Memory." *Personality and Social Psychology Bulletin* 21, 384-393.

Lipnack, J., and Stamps, J. 1997. *Virtual Teams: Reaching across Space, Time, and Organizations with Technology.* New York: Wiley.

Malone, T. W., and Crowston, K. 1990. "What Is Coordination Theory and How Can It Help Design Cooperative Work Systems?" In *Proceedings of CSCW '90,* 357-370.

Markus, M. L., Manville, B., and Agres, C. E. 2000. "What Makes a Virtual Organization Work?" *Sloan Management Review* 42, 1, 13-26.

Massey, A. P., Montoya-Weiss, M., and Hung, Y. C. 2002. "Synchronizing Pace in Asynchronous Global Virtual Project Teams." In *Proceedings of the Thirty-Fifth Hawaii International Conference on System Sciences.*

Massey, A. P., Montoya-Weiss, M. M., and Hung, Y. T. 2003. "Because Time Matters: Temporal Coordination in Global Virtual Project Teams." *Journal of Management Information Systems* 19, 4, 129-155.

Maznevski, M. L., and Chudoba, K. M. 2000. "Bridging Space over Time: Global Virtual Team Dynamics and Effectiveness." Pp. 473-492 in *Organization Science.* Linthicum.

McGrath, J. E. 1990. "Time Matters in Groups." In J. Galegher, R. E. Kraut and C. Egido (eds.) *Intellectual Teamwork: Social and Technical Foundations of Cooperative Work.* Hillsdale, NJ: Erlbaum, 23-61.

McGrath, J. E. 1991. "Time, Interaction, and Performance (Tip): A Theory of Groups." *Small Group Research* 22, 147-174.

Miller, C. R. 1984. "Genre as Social Action." *Quarterly Journal of Speech* 70, 151-167.

Moon, J. Y., and Sproull, L. 2002. "Essence of Distributed Work: The Case of the Linux Kernel." In P. Hinds and S. Kiesler (eds.) *Distributed Work.* Cambridge, MA: MIT Press, 381-404.

Mumford, L. 1934. *Technics and Civilisation.* New York: Harcourt, Brace & World.

Nippert-Eng, C. E. 1996. *Home and Work : Negotiating Boundaries through Everyday Life*. Chicago, IL: University of Chicago Press.

Nowotny, H. 1992. "Time and Social Theory: Towards a Social Theory of Time." *Time & Society* 1, 3, 421-454.

Ocker, R., Hiltz, S., Turoff, M., and J., F. 1995/1996. "The Effects of Distributed Group Support and Process Structuring on Software Requirements Development Teams: Results on Creativity and Quality." *Journal of Management Information Systems* 12, 3, 127-53.

Okhuysen, G. A., and Eisenhardt, K. M. 2002. "Integrating Knowledge in Groups: How Formal Interventions Enable Flexibility." *Organization Science* 13, 4, 370-386.

O'Leary, M. B., and Cummings, J. N. 2004. "Geographic Dispersion in Teams: The Interplay of Theory and Methods." Cambridge, MA.

O'Leary, M. B., Orlikowski, W., and Yates, J. 2002. "Distributed Work over the Centuries: Trust and Control in the Hudson's Bay Company." In P. Hinds and S. Kiesler (eds.) *Distributed Work*. Cambridge, MA: MIT Press, 27-54.

Olson, J. S., Teasley, S., Covi, L., and Olson, G. 2002. "The (Currently) Unique Advantages of Collocated Work." In P. Hinds and S. Kiesler (eds.) *Distributed Work*. Cambridge, MA: MIT Press, 113-135.

Orlikowski, W. J., and Yates, J. 1994. *Genre Repertoire : Norms and Forms for Work and Interaction*. Cambridge, MA: Alfred P. Sloan School of Management Massachusetts Institute of Technology.

Orlikowski, W. J., and Yates, J. 2002. "It's About Time: Temporal Structuring in Organizations." *Organization Science* 13, 6, 684-700.

Orlikowski, W. J., Yates, J., Fonstad, N. O., and Sloan School of Management. 2000. *Sloan 2001 : A Virtual Odyssey*. Cambridge, MA: Alfred P. Sloan School of Management Massachusetts Institute of Technology.

Orr, J. E. 1996. *Talking About Machines : An Ethnography of a Modern Job*. Ithaca, NY: ILR Press.

Owen, J., D. 1979. *Working Hours*. Lexington, MA: Lexington Books.

Panteli, N. 2004. "Articulations of Presence in Virtual Organizing." *Information and Organization* 14, 1, 59-81.

Pentland, B. 1995. " Information Systems and Organizational Learning: The Social Epistemology of Organizational Knowledge Systems." *Accounting, Management and Information Technologies* 5, 1-21.

Perlow, L. A. 1997. *Finding Time : How Corporations, Individuals, and Families Can Benefit from New Work Practices.* Ithaca, NY: ILR Press.

Perlow, L. A. 1998. "Boundary Control: The Social Ordering of Work and Family Time in a High-Tech Corporation." *Administrative Science Quarterly* 43, 2, 328-357.

Perlow, L. A. 1999. "The Time Famine: Toward a Sociology of Work Time." *Administrative Science Quarterly* 44, 1, 57-81.

Perlow, L. A. 2000. "Challenges for Work and Family in the Twenty-First Century." *Administrative Science Quarterly* 45, 3, 630-633.

Perlow, L. A. 2001. "Time to Coordinate - toward an Understanding of Work-Time Standards and Norms in a Multicountry Study of Software Engineers." *Work and Occupations* 28, 1, 91-111.

Pink, D. H. 2001. *Free Agent Nation: How America's New Independent Workers Are Transforming the Way We Live.* New York: Warner Business Books.

Rasker, P. C., Post, W. M., and Schraagen, J. M. C. 2000. "Effects of Two Types of Intrateam Feedback on Developing a Shared Mental Model in Command Con Control Teams." *Ergonomics* 43, 1167-1189.

Reddy, M., and Dourish, P. 2002. "A Finger on the Pulse: Temporal Rhythms and Information Seeking in Medical Work." In *Proceedings of the 2002 ACM Conference on Computer-Supported Cooperative Work*, 344-353.

Roy, D. F. 1960. "Banana Time: Job Satisfaction and Informal Interaction." In J Hassard (ed.) *Human Organizations.* London: Macmillan, 156-168. Reprinted as Chapter 9 in J. Hassard (ed.). (1990), The Sociology of Time.

Ruhleder, K., and Jordan, B. 1999. "Meaning-Making across Remote Sites: How Delays in Transmission Affect Interaction." In *Proceedings of the 6th European Conference on CSCW*, 411-429.

Schwarz, H. J. 2003. "Techno-Territories : The Spatial, Technological and Social Reorganization of Office Work." in *Program in Science, Technology and Society.* Cambridge, MA: MIT.

Seifert, C., and Hutchins, E. L. 1992. "Error as Oportunity: Learning in a Cooperative Task." *Human-Computer Interaction* 7, 4, 409-436.

Serfaty, D., Entin, E. E., and Johnston, J. H. 1998. "Team Coordination Training." In J. A. Cannon-Bowers and E. Salas (eds.) *Making Decisions under Stress: Implications for Individual and Team Training.* Washington, DC: American Psychological Association, 221-246.

Sheppard, B. H., and Sherman, D. M. 1998. "The Grammars of Trust: A Model and General Implications." *Academy of Management Review* 23, 422-437.

Soloman, C. 2001. "Managing Virtual Teams." *Workforce* 80, 6, 60-64.

Sorokin, P. A., and Merton, R. A. 1937. "Social Time: A Methodological and Functional Analysis." *American Journal of Sociology* 42, 615-629.

Sproull, L., and Kiesler, S. 1986. "Reducing Social Context Cues: Electronic Mail in Organizational Communication." *Management Science* 32, 1492-1512.

Sproull, L., and Kiesler, S. 1991. "Computers, Networks and Work." *Scientific American* 265, 3, 116-123.

Staudenmayer, N., Tyre, M., and Perlow, L. 2002. "Time to Change: Temporal Shifts as Enablers of Organizational Change." *Organization Science* 13, 5, 583-597.

Steward, B. 2000. "Changing Times: The Meaning, Measurement and Use of Time in Teleworking." *Time & Society* 9, 1, 57-74.

Straus, S. G., and McGrath, J. E. 1994. "Does the Medium Matter? The Interaction of Task Type and Technology on Group Performance and Member Reactions." *Journal of Applied Psychology* 79, 87-97.

Struss, A., and Corbin, J. 1997. *Basics of Qualitative Research: Grounded Theory Procedures and Techniques*. Newbury Park, CA: Sage.

Suarez, F., Cusumano, M., and Fine, C. 1995. "An Empirical Study of Flexibility in Manufacturing." *Sloan Management Review*, Fall, 25-32.

Suchman, L. A. 1987. *Plans and Situated Actions : The Problem of Human-Machine Communication*. Cambridge / New York: Cambridge University Press.

Tausig, M., and Fenwick, R. 2001. "Unbinding Time: Alternate Work Schedules and Work-Life Balance." *Journal of Family and Economic Issues* 22, 101-119.

Thompson, E. P. 1967. "Time, Work-Discipline and Industrial Capitalism." *Past and Present* 38, 56-97.

Tietze, S., and Mussion, G. 2002. "When 'Work' Meets 'Home': Temporal Flexibility as Lived Experience." *Time & Society* 11, 2/3, 315-334.

Townsend, A. M., DeMarie, S. M., and Hendrickson, A. R. 1998. "Virtual Teams: Technology and the Workplace of the Future." *Academy of Management Executive* 12, 17-29.

Walther, J. B. 2002. "Time Effects in Computer-Mediated Groups: Past, Present, and Future." In P. Hinds and S. Kiesler (eds.) *Distributed Work.* Cambridge, MA: MIT Press, 235-257.

Weick, K., and Roberts, K. 1993. "Collective Mind in Organizations: Heedful Interrelating on Light Decks." *Administrative Science Quarterly* 38, 3, 357-381.

Weisband, S. 2002. "Maintaining Awareness in Distributed Team Collaboration: Implications for Leadership and Performance." In P. Hinds and S. Kiesler (eds.) *Distributed Work.* Cambridge, MA: MIT Press, 311-334.

Whittaker, S., and Schwarz, H. 1999. "Meetings of the Board: The Impact of Scheduling Medium on Long Term Group Coordination in Software Development." *Computer Supported Cooperative Work* 8, 175-205.

Whyte, W. H. 1956. *The Organization Man.* New York: Simon and Schuster.

Xu, H., Yates, J., and Orlikowski, W. J. 2005. "Who's Doing What, When?: Coordinating Work in a Distributed Software Development Team." In *The 2005 Academy of Management Conference in Honolulu.*

Yates, J., and Orlikowski, W. 1992. "Genres of Organizational Communication: A Structurational Approach to Studying Communication and Media." A*cademy of Management Review* 17, 2, 299-326.

Yates, J., and Orlikowski, W. J. 2001. "Genre Systems: Chronos and Kairos in Communicative Interaction." In R. Coe, L. Lingard and T. Teslenko (eds.) T*he Rhetoric and Ideology of Genre: Strategies for Stability and Change.* Cresskill, NJ: Hampton, 103-121.

Yates, J., and Orlikowski, W. J. 2002. "Genre Systems: Structuring Interactions through Communicative Norms." J*ournal of Business Communication* 39, 1, 13-35.

Yates, J., Orlikowski, W. J., and Fonstad, N. 2001. "Sloan 2001: A Virtual Odyssey." In I. Zigurs and L. Chidambaram (eds.) O*ur Virtual World: The Transformation of Work, Play, and Life Via Technology.* Hershey, PA: Idea Group Publishing.

Yates, J., Orlikowski, W. J., and Okamura, K. 1999. "Explicit and Implicit Structuring of Genres: In Electronic Communication: Reinforcement and Change in Social Interaction." O*rganization Science* 10, 1, 83-103.

Yates, J., Orlikowski, W. J., Woerner, S. L., and Sloan School of Management. 2003. V*irtual Organizing : Using Threads to Coordinate Distributed Work.* Cambridge, MA: MIT Sloan School of Management.

Yoo, Y., and Alavi, M. 2004. "Emergent Leadership in Virtual Teams: What Do Emergent Leaders Do?" I*nformation and Organization* 14, 1, 27-58.

Zerubavel, E. 1979. P*atterns of Time in Hospital Life : A Sociological Perspective.* Chicago, IL: University of Chicago Press.

Zerubavel, E. 1981. H*idden Rhythms : Schedules and Calendars in Social Life.* Chicago, IL: University of Chicago Press.

Zerubavel, E. 1985. T*he Seven Day Circle : The History and Meaning of the Week.* New York/London: Free Press/Collier Macmillan.

Appendix A. Coding scheme I

CATEGORIES	DEFINITION	PERCENTAGE	
		Phase I†	Phase II‡
A. Why (Purpose)	Purpose of message		
A1 Response	Reply to previous message(s)	44.2	46.4
A2 Query/Solicitation	Ask question(s) or request ideas, information, or participation	14.5	10.4
A3 Scheduling	Scheduling meetings or events	2.6	2.8
A4 Coordination	Coordinate tasks or work between members	10.3	8.2
A5 Report	Document an event, trip, or meeting	1.4	1.0
A6 Status Report	Report the status or progress of work	9.4	3.4
A7 Update Notification	Notify an update on server, web, etc.	9.4	16.4
A8 Bug/Error Notification	Notify bug(s) or error(s)	8.1	4.0
A9 Announcement	Indicate an event or change in affairs	4.9	10.3
A10 FYI	Offer information	4.1	8.3
A11 Proposal	Propose ideas, solutions, procedures, projects	9.4	9.0
A12 Request	Request something to other people	7.8	9.2
A13 Discussion	Discuss a specific topic	32.1	19.7
A14 Other	Residual purpose category	(N/A)	(N/A)
B. What (Content)	Content of message		
B1 Work related	Projects, product, or company	96.0	91.3
B1.1 Technical	Technical aspects of product or technology	82.9	45.6
B1.2 Administrative	Running of the company	81.4	51.7
B2 Not work But Relevant	Not directly deal with, but is relevant to work	3.8	7.3
B3 Personal	Personal matters	3.0	3.5
B4 Other	Residual content category	(N/A)	(N/A)
C. Who/m	Sender and recipients of message		
C1 To	Recipient(s) of message	(N/A)	(N/A)
C2 From	Sender of message	(N/A)	(N/A)
C3 CC	Recipient(s) of the copy of a message	(N/A)	(N/A)
C4 FWD from	Sender of an original message that is forwarded	(N/A)	(N/A)
D. Where	Any reference to the location of members or work		
D1 Reference to Travel	Any ref. to a travel	1.0	1.9
D2 Reference to Physical Space	Any ref. to physical locations of members or work	4.4	4.8
D3 Reference to Virtual Space	Any ref. to virtual space	22.3	19.1
E. How (Form)	Format of message		
E1 Code	Presence of code	11.3	20.4
E2.1 Dialogue	Previous message(s) edited before inclusion	10.3	12.5
E2.2 Mosaic	Previous message(s) unedited before inclusion	15.5	18.5
E3 Subject Line Only	No body content or signature/attachment/code/cvs log only	0.7	4.9

E4 Standardized	Format based on a standardized protocol	1.4	13.5
E5 Machine-generated	Generated and sent automatically by the system	0.0	0.1
E6 Other	Residual format category	(N/A)	(N/A)
F. When	Indicates temporal patterns of work and communication		
F1. Time Stamp	Date, time, and day of week of message	(N/A)	(N/A)
F2. Temporal Reference	Indicates the instances of specific temporal practices		
F2.1 External Event-based	Indicates temporal rhythm based on external events	1.5	1.9
F2.2 Internal Event-based	Indicates temporal rhythm based on internal events	7.8	4.9
F2.3 Clock-based	Indicates temporal rhythm based on clock or calendar	11.6	12.6
F2.4 Work/Personal life	Refer to work time and family time	2.6	3.0
F2.5 Sequencing	Indicates sequence of things to do for a task	8.9	3.1
F3 Other	Residual temporal practice category	(N/A)	(N/A)
G. Work Practice	Indicates the work practices		
G1.1 Reference to phone call	Refer to or include the content of phone conversations	1.2	6.2
G1.2 Reference to face to face meeting	Refer to or include the content of face-to-face meetings	0.4	0.7
G2 Suggest phone call	Suggest phone calls or phone meetings	1.2	2.6
G3 Reference to problems	Reference to problems (ex. Technical, procedural, communicational...)	23.1	14.7
G4 Reference to opportunities	Indicates opportunities associated with technology or business etc.	1.0	5.1
G5 Establishing conventions	Attempts to set a rule regarding a specific work practice	0.6	2.6
G6 Geographical dispersion	Indicates issues, problems, concerns or benefits of distributed work	0.6	1.7
G7 Boundary management	Indicates practices of creating, maintaining, negotiating, and managing boundaries between work/home, work/family etc.	1.8	4.5
G8 Requesting help	Asks help from other members to solve one's problem encountered doing one's work	5.9	5.6
G9 Offering help	Tries to help out other members' work (or problems)	4.8	4.6
G10 Disagreement	Indicates any kind of disagreement among members. (e.g., over technical decisions, work procedures, personnel hire.)	2.4	2.8
G11 Reference to external expertise or network	Refers to external expert or network for information, collaboration, or outsourcing (e.g. soliciting or working with external experts on various tasks of product development (technical, marketing, licensing, financing...)	0.9	5.7
G12 Others	Residual work practices category	(N/A)	(N/A)

† This percentage is out of the 3,518 email messages from March 1997 to February 1998. As most categories are not mutually exclusive, percentages do not add up to 100%.

‡ This percentage is out of the 3,009 email messages from April 1999 to March 2000. As most categories are not mutually exclusive, percentages do not add up to 100%.

Appendix B. Coding scheme II

CATEGORIES	DEFINITION	PERCENTAGE †
0. Identification	Message ID of a message	(N/A)
0.1. Already coded	A duplicate or a copy of an original message	(N/A)
A. Why (Purpose)		
A1.Initiating discussion	A message that initiates a discussion (work related) among member, for example, by posing a question, or announcing a fact or change, or providing arguments, or proposing a plan or agenda, or requesting something.	31.2
A2.Coordinating/scheduling	A message to coordinate work (tasks) among members or to schedule meetings or events. NOTE: Status Report belongs to this category.	11.0
A3.Discussion	A follow-up message or response to a message of category A1. But the discussion should be relevant to the work.	37.0
A4. Report	A message to report a meeting, an event, or a fact. (e.g., meeting report (internal and external), expense report.)	0.4
A5. Notification	Notifying a change already happened or tasks completed or bugs found. Ex. Update notification, bug/error notification.	19.5
A6. Other	A message that does not fit into the current categories. Also make the categories as exclusive as possible, and try to code a message under one purpose category.	(N/A)
B. What		
B1. Technical	Content related to the technical aspects of product or technology	60.1
B2. Administrative	Content related to the running of the company	42.9
B2.1. Administrative What	Description of the administrative issue	(N/A)
B3.Personal	Personal content	5.1
B4. Other	A message that does not fit into the current categories.	(N/A)
C. When		
C1.Internal phasing	Reference to internal phasing for tasks. Ex. Deadlines and milestones set by members.	9.1
C1.1. Clock-based	Internal task-phasing based on specific times.	7.1
C1.2. Event-based	Internal task-phasing based on specific events.	1.5
C2.External deadlines	Reference to external phasing for tasks, Ex. Upcoming conference, or marketing visit, or market condition.	1.8
C2.1. Clock-based	External task-phasing based on specific times.	1.5
C2.2. Event-based	External task-phasing based on specific events.	0.1
C3. Other	A message that does not fit into the current categories.	(N/A)
D. Work Practice		
D1.Reference to media	Incidents of media switches between various	9.4

switches	communication media e.g. as suggesting phone calls, reference to face-to-face or phone conversations.	
D2. Instant messaging	A message that has only the subject line (or a signature in the body content).	1.3
D3.Reference to problems	Reference to problems (ex. Technical, procedural, communicational...)	23.7
D4.Requesting help	A message explicitly asking help from other members to solve one's problem encountered doing one's work	3.5
D5.Offering help	A message trying to help out other members' work (or problems)	3.0
D6.Disagreement	Incidents of any kind of disagreement among members. (e.g., over technical decisions, work procedures, personnel hire.)	2.4
D7.Reference to external expertise/network	Reference to external expert network for collaboration or outsourcing, e.g. soliciting or working with external experts on various tasks of product development (technical, marketing, licensing, financing...)	6.8
D8.Negotiation (internal)	Reference to any incidents of internal negotiations (e.g. on scheduling, deadlines, technical decisions, task assignments)	4.9
D8.1.Negotiating expertise	Negotiating each member's expertise to distribute tasks	0.9
D9. Identity	Reference to any personal or organizational identities.	1.4
D9. What	Description of the identity (s)	(N/A)
D10. Other	A message that does not fit into the current categories.	(N/A)

† This percentage is out of the total 11,803 email messages from July 1996 to December 2000. As most categories are not mutually exclusive, percentages do not add up to 100%.

Appendix C: Dan's home office in the basement

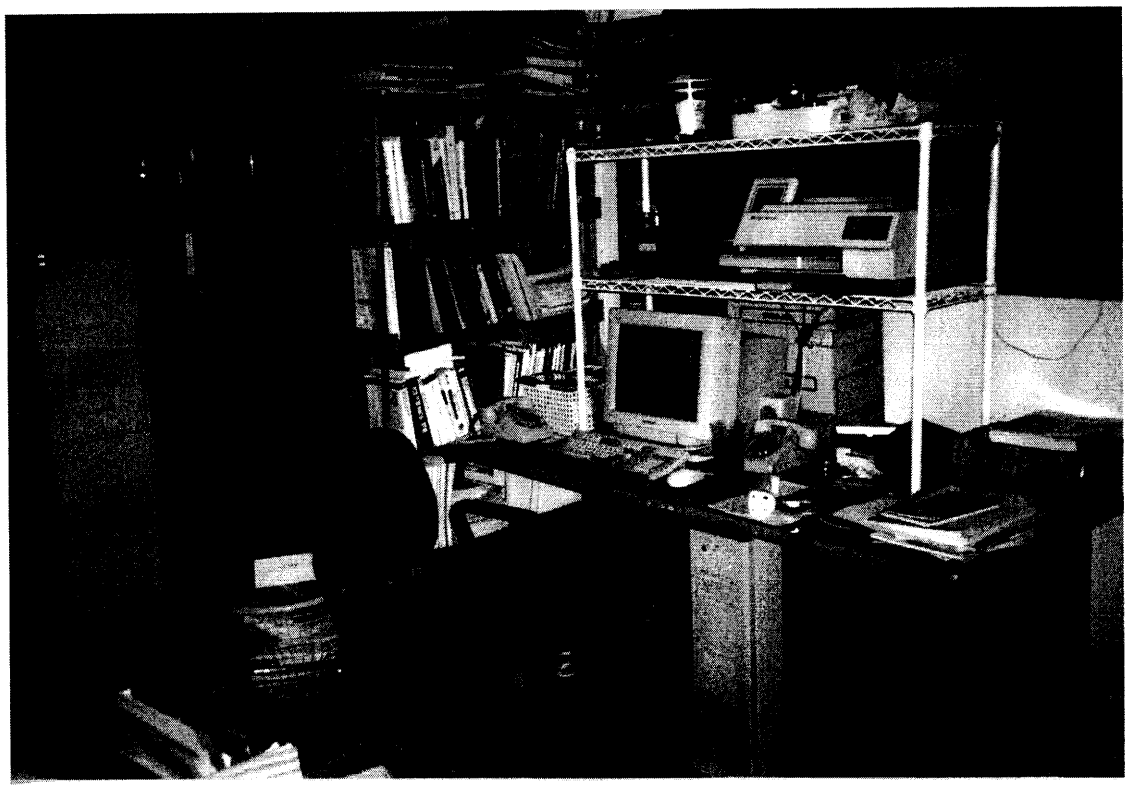


Photo courtesy of LC

Appendix D: Robert's home office

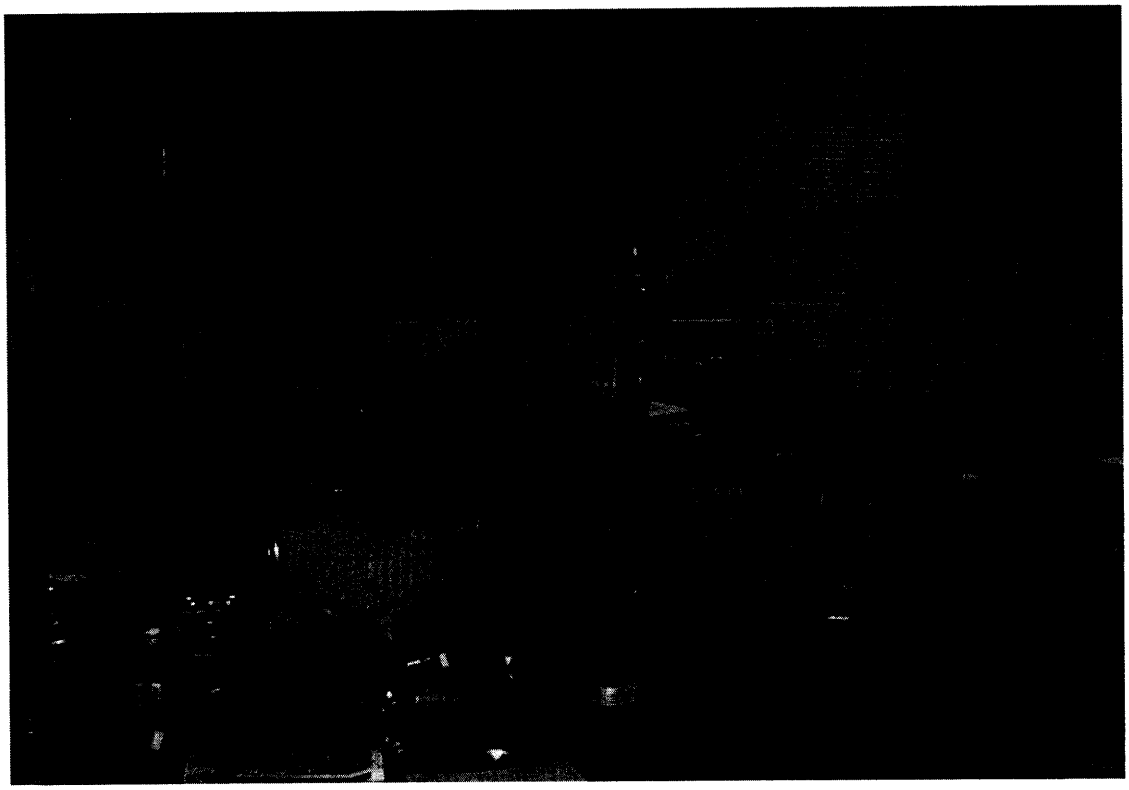


Photo courtesy of LC